

**Oracle<sup>®</sup> Retail Merchandising System  
Operations Guide – Volume 1  
Batch Overviews and Designs  
Release 12.0  
May 2006**

Copyright © 2006, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software—Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

# Contents

|                                                         |           |
|---------------------------------------------------------|-----------|
| <b>Preface</b>                                          | <b>xi</b> |
| Audience                                                | xi        |
| Related Documents                                       | xi        |
| Customer Support                                        | xi        |
| <b>1 Introduction</b>                                   | <b>1</b>  |
| What's in the guide                                     | 1         |
| Volume 1 – Batch Overviews and Designs                  | 1         |
| Volume 2 – Message Publication and Subscription Designs | 1         |
| Volume 3 – Backend Configuration and Operations         | 2         |
| RMS Modules                                             | 2         |
| The SYSTEMS_OPTIONS Table                               | 2         |
| <b>2 Audit Trail Batch</b>                              | <b>3</b>  |
| Overview                                                | 3         |
| Batch Designs Summary                                   | 3         |
| auditprg (Audit Purge Process)                          | 3         |
| auditsys (Audit Logic Information Edits)                | 4         |
| <b>3 Calendar Batch</b>                                 | <b>7</b>  |
| Overview                                                | 7         |
| Batch Design Summary                                    | 7         |
| dtesys (Increment Set System Date)                      | 7         |
| <b>4 Competitive Pricing Batch</b>                      | <b>9</b>  |
| Overview                                                | 9         |
| Competitive Price Change Process                        | 9         |
| Batch Design Summary                                    | 9         |
| cmpprg.pc (Competitive Pricing Purge)                   | 9         |
| cmpupld (Competitive Pricing Upload)                    | 11        |
| <b>5 Contracts Batch</b>                                | <b>15</b> |
| Overview                                                | 15        |
| Batch Design Summary                                    | 15        |
| cntrmain (Contract Maintenance and Purging)             | 16        |
| cntrordb (Contract Replenishment Type 'B')              | 17        |
| <b>6 Cost Change Batch</b>                              | <b>19</b> |
| Overview                                                | 19        |
| Cost Change Process                                     | 19        |
| Multi-channel Supplier Cost Change Rules:               | 19        |
| Batch Design Summary                                    | 20        |
| ccprg (Cost Event Purge)                                | 20        |
| sccect (Supplier Cost Change Extract)                   | 22        |
| <b>7 Daily Purge Batch</b>                              | <b>24</b> |
| Overview                                                | 24        |
| Batch Design Summary                                    | 24        |
| dlyprg (Daily Purge)                                    | 24        |

|                                                                                          |            |
|------------------------------------------------------------------------------------------|------------|
| <b>8 Deals Maintenance Batch .....</b>                                                   | <b>30</b>  |
| Overview.....                                                                            | 30         |
| Deal Concepts.....                                                                       | 30         |
| Rules that Apply to Deal Functionality .....                                             | 31         |
| Deal Process for Off-invoice .....                                                       | 31         |
| Deal Process for Bill-backs.....                                                         | 32         |
| Deals Process.....                                                                       | 32         |
| Batch Design Summary .....                                                               | 34         |
| costcalc (Deal Cost Calculation).....                                                    | 35         |
| dealact (Deal Actuals).....                                                              | 37         |
| dealcls (Deal Close).....                                                                | 38         |
| dealday (Deal Tran Data Extract Summed Daily) .....                                      | 39         |
| dealex (Deal Explode).....                                                               | 40         |
| dealfct (Deal Forecast).....                                                             | 42         |
| dealinc (Deal Income Calculation) .....                                                  | 43         |
| dealprg (Deals Purge) .....                                                              | 44         |
| dealupld (Deal Upload).....                                                              | 46         |
| discothapply (Discount OTB Apply) .....                                                  | 70         |
| ditinsrt (Deal Item Insert).....                                                         | 72         |
| orddscnt (Order Deal Discount).....                                                      | 74         |
| precostcalc (Pre-cost Calculation) .....                                                 | 75         |
| vendinvc (Vendor Deal Invoicing) .....                                                   | 77         |
| vendinvf (Vendor Deal Invoicing).....                                                    | 79         |
| <b>9 Diff Ratio Build Batch.....</b>                                                     | <b>81</b>  |
| Overview.....                                                                            | 81         |
| Batch Design Summary .....                                                               | 81         |
| dfrtbld (Diff Ratio Build).....                                                          | 81         |
| <b>10 Electronic Data Interchange (EDI) Batch .....</b>                                  | <b>83</b>  |
| Overview.....                                                                            | 83         |
| RMS files and EDI Translations.....                                                      | 83         |
| Batch Design Summary .....                                                               | 83         |
| edidladd (EDI Location Address to Vendor Download).....                                  | 83         |
| edidlcon (EDI Contract Information Downloads).....                                       | 86         |
| edidlord (EDI purchase order download).....                                              | 89         |
| edidlprd (EDI Sales and Stock On Hand Report Download).....                              | 98         |
| ediprg (EDI Purge).....                                                                  | 102        |
| ediupack (EDI Supplier Order Acknowledgements and Changes) .....                         | 103        |
| ediupadd (EDI Supplier Address Upload) .....                                             | 107        |
| ediupavl (Supplier Availability for Contracts Upload) .....                              | 109        |
| ediupcat (New and Changed Upload from Supplier).....                                     | 111        |
| <b>11 Forecasting Batch .....</b>                                                        | <b>121</b> |
| Overview.....                                                                            | 121        |
| Batch Design Summary .....                                                               | 121        |
| fcstprg (Oracle Retail Demand Forecasting Purge) .....                                   | 121        |
| fcstrbld (Oracle Retail Demand Forecasting Rollup).....                                  | 122        |
| fcstrbld_sbc (Oracle Retail Demand Forecasting Rollup by Department, Class and Subclass) |            |
| .....                                                                                    | 124        |
| ftmednld (Time Hierarchy Download).....                                                  | 125        |
| soutdnld (Stockout Download) .....                                                       | 126        |



|                                                          |            |
|----------------------------------------------------------|------------|
| <b>12 General Ledger (GL) Batch .....</b>                | <b>129</b> |
| Overview.....                                            | 129        |
| Batch Design Summary .....                               | 129        |
| dealfinc (Deal Fixed Income) .....                       | 129        |
| fifglnd1 (General Ledger Interface).....                 | 130        |
| fifglnd2 (General Ledger Interface).....                 | 132        |
| fifglnd3 (General Ledger Interface).....                 | 133        |
| <b>13 Geocode Hierarchy Batch.....</b>                   | <b>135</b> |
| Overview.....                                            | 135        |
| Batch Design Summary .....                               | 135        |
| gcupld (Geocode Hierarchy Upload).....                   | 135        |
| <b>14 Inventory Adjustment Batch.....</b>                | <b>139</b> |
| Overview.....                                            | 139        |
| Batch Design Summary .....                               | 139        |
| invaprg (Inventory Adjustment Purge) .....               | 139        |
| <b>15 Invoice Matching Batch .....</b>                   | <b>141</b> |
| Overview.....                                            | 141        |
| Batch Design Summary .....                               | 141        |
| edidlinv (EDI Invoice Download).....                     | 141        |
| invclshp (Invoice Close Shipments) .....                 | 148        |
| invprg (Invoice Purge) .....                             | 149        |
| <b>16 Open to Buy Maintenance Batch.....</b>             | <b>152</b> |
| Overview.....                                            | 152        |
| Batch Design Summary .....                               | 153        |
| onictext (On Inter-company Transfer Exhibit).....        | 153        |
| onorddnld (On Order Download to Financial Planning)..... | 154        |
| onordext (On Order Extract) .....                        | 156        |
| otbdlord (Outstanding Order Export Download) .....       | 158        |
| otbdlsal (Open To Buy Download Stock Ledger).....        | 160        |
| otbdnld (Open To Buy Download) .....                     | 166        |
| otbprg (Open to Buy Purge).....                          | 170        |
| otbupfwd (Accept Forward Limit Percentages Upload) ..... | 171        |
| otbupld (New Budget Data and Budget Adjustments).....    | 173        |
| stlgdnld (Stock Ledger Download) .....                   | 176        |

|                                                                  |            |
|------------------------------------------------------------------|------------|
| <b>17 Oracle Retail Sales Audit Batch .....</b>                  | <b>181</b> |
| Overview.....                                                    | 181        |
| Store Day Defined .....                                          | 181        |
| Making Changes in the CODE_DETAIL Table .....                    | 181        |
| Preparation for the Data Import .....                            | 182        |
| ReSA's Conversion from the Selling UOM to the Standard UOM ..... | 183        |
| A Note about Primary Variant Relationships.....                  | 184        |
| Transaction Data Import and Validation.....                      | 185        |
| Trickle Polling .....                                            | 185        |
| The DCLOSE Transaction Type.....                                 | 186        |
| Total Calculations and Rules .....                               | 187        |
| Export Store Day Transaction Data to Applications .....          | 187        |
| Transaction Data Exports and the Unit of Work.....               | 188        |
| Oracle Retail Merchandise System (RMS) Export .....              | 189        |
| Oracle Retail Data Warehouse (RDW) export.....                   | 189        |
| Account Clearing House (ACH) Export .....                        | 189        |
| Universal Account Reconciliation System (UAR) Export.....        | 189        |
| Oracle Retail Invoice Matching (ReIM) Export .....               | 190        |
| Escheatment Totals to ReIM for Accounts Payable .....            | 190        |
| Full Disclosure and Post-export Changes .....                    | 190        |
| What Happens to Totals when Transactions are Modified?.....      | 190        |
| Adjustments Received from an Application .....                   | 191        |
| Oracle Retail Sales Audit Dataflow Diagrams.....                 | 191        |
| Batch Design Summary of ReSA Modules.....                        | 192        |
| saescheat (Sales Audit Escheated Vouchers).....                  | 193        |
| saexpach (Sales Audit ACH Download).....                         | 194        |
| saexpagl (Sales Audit Export to GL).....                         | 201        |
| saexpim (Sales Audit Export to Invoice Matching) .....           | 203        |
| saexprdw (Sales Audit Export to RDW).....                        | 205        |
| Oracle Retail Sales Audit (ReSA) – File Layout – RDW .....       | 207        |
| saexprms (Sales Audit Export to RMS).....                        | 233        |
| saexpuar (Universal Account Reconciliation System Export) .....  | 238        |
| sagetref (Sales Audit Get Reference).....                        | 241        |
| saimpadj (Sales Audit Import Adjustments) .....                  | 246        |
| saimptlog (Sales Audit Import).....                              | 248        |
| ReSA Interface File Layout [rtlog] .....                         | 267        |
| saimptlogfin (Sales Audit Import) .....                          | 288        |
| sapreexp (Sales Audit Pre-Export).....                           | 289        |
| sapurge (Sales audit Purge).....                                 | 291        |
| sarules (Sales Audit Rules) .....                                | 294        |
| sastdyocr (Sales Audit Store Day Create).....                    | 295        |
| satotals (Sales Audit Totals) .....                              | 297        |
| savouch (Sales Audit Voucher Upload).....                        | 299        |

|                                                                               |            |
|-------------------------------------------------------------------------------|------------|
| <b>18 Oracle Retail Trade Management Batch .....</b>                          | <b>303</b> |
| Overview.....                                                                 | 303        |
| Invoice and Accounts Payable Integration.....                                 | 303        |
| Batch Design Summary .....                                                    | 304        |
| Letter of Credit .....                                                        | 304        |
| cednld (Customs Entry Download).....                                          | 304        |
| htsupld (Harmonized Tariff Schedule Upload).....                              | 312        |
| lcanld (Letter of Credit Application Download).....                           | 323        |
| lcmdnld (Letter of Credit Amendment Download).....                            | 338        |
| lcmt700 (SWIFT File Conversion) .....                                         | 344        |
| lcmt707 (Converts Letter of Credit from Oracle Retail Format to SWIFT) .....  | 346        |
| lcmt730 (SWIFT File Conversion - Letter of Credit Confirmation).....          | 349        |
| lcmt798 (SWIFT File Conversion – Letter of Credit Charges and Drawdowns)..... | 353        |
| lcup798 (Letter of Credit Update).....                                        | 361        |
| lcupld (Letter of Credit Upload) .....                                        | 364        |
| tranupld (Transportation Upload) .....                                        | 366        |
| <b>19 Organization Hierarchy Batch .....</b>                                  | <b>373</b> |
| Overview.....                                                                 | 373        |
| Organization Hierarchy Concepts.....                                          | 373        |
| Location Retail Overview .....                                                | 374        |
| Batch Design Summary .....                                                    | 374        |
| lclrbld (Location List Batch Rebuild) .....                                   | 374        |
| likestore (Store Add ‘Like Store’ Processing) .....                           | 375        |
| schedprg (Store Ship Schedule Purge).....                                     | 377        |
| storeadd (Store Add).....                                                     | 378        |
| whadd (Warehouse Add) .....                                                   | 379        |
| whstrasg (Warehouse Store Assignment) .....                                   | 381        |
| <b>20 Point-of-Sale (POS) Download Batch .....</b>                            | <b>383</b> |
| Overview.....                                                                 | 383        |
| Point of Sale Download .....                                                  | 384        |
| Batch Design Summary .....                                                    | 384        |
| poscdnld (Point Of Sale Configuration Download) .....                         | 384        |
| posdnld (Point Of Sale Download) .....                                        | 395        |
| posgpdlld (Group Number and Department Number Download to POS).....           | 405        |
| <b>21 Prepost Batch .....</b>                                                 | <b>408</b> |
| Overview.....                                                                 | 408        |
| Batch Design Summary .....                                                    | 408        |
| prepost (Pre/Post).....                                                       | 408        |
| <b>22 Pricing Interface to Oracle Retail Price Management (RPM) .....</b>     | <b>417</b> |
| Overview.....                                                                 | 417        |
| Batch Design Summary .....                                                    | 417        |
| rpmmovavg (RPM Moving Average) .....                                          | 417        |
| <b>23 Purchase Order Batch.....</b>                                           | <b>419</b> |
| Overview.....                                                                 | 419        |
| Batch Design Summary .....                                                    | 419        |
| genpreiss (Pre-Issued Order Number Generation) .....                          | 419        |
| ordautcl (PO Auto Close).....                                                 | 420        |
| ordprg (Order Purge).....                                                     | 423        |
| ordrev (Purchase Order Information Written to Order History Table) .....      | 428        |
| ordupd (Order Update).....                                                    | 429        |
| vrplbld (Vendor replenished order build) .....                                | 430        |

|                                                                                        |            |
|----------------------------------------------------------------------------------------|------------|
| <b>24Reclassification Batch .....</b>                                                  | <b>433</b> |
| Overview.....                                                                          | 433        |
| Batch Design Summary .....                                                             | 433        |
| cremhierdly (Create Merchandise Hierarchy Daily) .....                                 | 433        |
| reclsdly (Reclassification of Item) .....                                              | 435        |
| <b>25Replenishment Batch.....</b>                                                      | <b>437</b> |
| Overview.....                                                                          | 437        |
| Replenishment Process .....                                                            | 437        |
| Code Values.....                                                                       | 437        |
| Investment Buy .....                                                                   | 437        |
| Investment Buy System Options.....                                                     | 438        |
| Batch Design Summary .....                                                             | 439        |
| cntrprss (Contract Replenishment).....                                                 | 439        |
| ibcalc (Investment Buy Calculation).....                                               | 441        |
| ibexpl (Investment Buy Explosion) .....                                                | 443        |
| ociroq (Recommended Order Quantity).....                                               | 445        |
| repladj (Replenishment Adjustment) .....                                               | 446        |
| reqext (Replenishment Quantity Extract).....                                           | 447        |
| rilmaint (Replenishment Item Location Maintenance) .....                               | 449        |
| rplapprv (Automatic Replenishment Order Approval) .....                                | 451        |
| rplatupd (Replenishment Attribute Update).....                                         | 453        |
| rplbld (Replenishment Order Build) .....                                               | 454        |
| rplext (Vendor Replenishment Extraction) .....                                         | 456        |
| rplprg (Replenishment Purge).....                                                      | 458        |
| rplprg_month (Replenishment Purge).....                                                | 459        |
| rplsplsplit (Replenishment/Truck Splitting) .....                                      | 461        |
| supcnstr (Supplier Constraint Scaling).....                                            | 463        |
| <b>26RETL Extraction Programs.....</b>                                                 | <b>465</b> |
| Overview.....                                                                          | 465        |
| RMS Extract Data (Based on RDW Dimension Data) .....                                   | 465        |
| RMS Extract Data (Based on RDW Fact Data) .....                                        | 470        |
| Maintenance Programs.....                                                              | 477        |
| Flow Diagrams and Legend .....                                                         | 478        |
| RETL API Flat File Specifications .....                                                | 490        |
| API Format .....                                                                       | 490        |
| File Layout.....                                                                       | 490        |
| General Business Rules and Standards Common to All APIs .....                          | 492        |
| <b>27Sales Posting Batch .....</b>                                                     | <b>657</b> |
| Overview.....                                                                          | 657        |
| The POS upload process.....                                                            | 657        |
| Processing POSU data .....                                                             | 658        |
| A Note about Oracle Retail Sales Audit and POSUPLD.PC .....                            | 662        |
| Batch Design Summary .....                                                             | 662        |
| hstbld (Sales History Rollup by Department, Class and Subclass).....                   | 662        |
| hstbld_diff (Sales History Rollup by Diff_IDs).....                                    | 664        |
| hstbldmth (Monthly Sales History Rollup By Department, Class And Subclass) .....       | 665        |
| hstbldmth_diff (Sales History Rollup By Diff Ids Per Month) .....                      | 666        |
| hstmhthupd (Monthly Stock on Hand, Retail and Average Cost Values Update) .....        | 668        |
| hstprg (Purge Sales History) .....                                                     | 669        |
| hstprg_diff (Sales History Purge by Diff).....                                         | 670        |
| hstwkupd (Weekly Stock on Hand and Retail Value Update for Fashion Item/Location) .... | 671        |
| posupld (Point Of Sales Upload) .....                                                  | 673        |

|                                                                 |            |
|-----------------------------------------------------------------|------------|
| <b>28 Scheduled Item Maintenance Batch .....</b>                | <b>681</b> |
| Overview.....                                                   | 681        |
| Security Feature for Item Lists.....                            | 681        |
| Batch Design Summary .....                                      | 681        |
| sitmain (Scheduled Item Maintenance Main) .....                 | 681        |
| <b>29 Stock Count Batch .....</b>                               | <b>683</b> |
| Overview.....                                                   | 683        |
| Stock Count Types: Units Versus Units and Monetary Values ..... | 684        |
| Stock Count – Unit Only .....                                   | 684        |
| Stock Count – Unit and Monetary Value.....                      | 684        |
| Stock Count Process .....                                       | 684        |
| Stock Count Request.....                                        | 685        |
| Batch Design Summary .....                                      | 685        |
| lifstkup (Stock Upload Conversion) .....                        | 685        |
| stkdlly (Stock Count Shrinkage Update).....                     | 689        |
| stkprg (Purge Stock Count).....                                 | 690        |
| stkschedxpld (Scheduled Stock Count Explode) .....              | 691        |
| stkupd (Stock Count Snapshot Update) .....                      | 693        |
| stkupld (Upload Stock Count).....                               | 694        |
| stkvar (Stock Count Stock on Hand Updates).....                 | 698        |
| stkxpld (Stock Count Explode).....                              | 699        |
| <b>30 Stock Ledger Batch.....</b>                               | <b>702</b> |
| Overview.....                                                   | 702        |
| Stock Ledger Set Up and Accounting Methods.....                 | 702        |
| Stock Counts and Budget Shrinkage.....                          | 705        |
| PL/SQL Packages .....                                           | 705        |
| End of Year (NWP) Inventory .....                               | 705        |
| Batch Design Summary .....                                      | 706        |
| nwppurge (End Of Year Inventory Position Purge).....            | 706        |
| nwpyearend (End of Year Inventory Position Snapshot).....       | 707        |
| salapnd (Stock Ledger Append).....                              | 708        |
| saldly (Sales Daily) .....                                      | 710        |
| saleoh (End of Half Stock Ledger Processing) .....              | 711        |
| salins (Stock Ledger and Budget Tables Insert).....             | 712        |
| salmaint (Stock Ledger Table Maintenance) .....                 | 714        |
| salmth (Monthly Stock Ledger Processing).....                   | 715        |
| salprg (Purge Stock Ledger Transactions) .....                  | 716        |
| salstage (Stock Ledger Stage).....                              | 717        |
| salweek (Sales Weekly) .....                                    | 719        |
| wasteadj (Wastage adjustment).....                              | 721        |
| <b>31 Stock Order Receipt Reconciliation Batch.....</b>         | <b>723</b> |
| Overview.....                                                   | 723        |
| Batch Design Summary .....                                      | 723        |
| dummyctn (Dummy Carton).....                                    | 723        |
| tamperctn (Tampered carton).....                                | 725        |
| <b>32 Store Grade Batch .....</b>                               | <b>727</b> |
| Overview.....                                                   | 727        |
| Batch Design Summary .....                                      | 727        |
| gradupld (Store Grade Upload).....                              | 727        |

|                                                        |            |
|--------------------------------------------------------|------------|
| <b>33 Supplier Batch .....</b>                         | <b>730</b> |
| Overview.....                                          | 730        |
| Batch Design Summary .....                             | 730        |
| supmth (Supplier Data Amount Repository).....          | 730        |
| <b>34 Tax Rate Batch.....</b>                          | <b>733</b> |
| Overview.....                                          | 733        |
| Batch Design Summary .....                             | 734        |
| tifposdn (Tax Rate POS Download) .....                 | 734        |
| txrposdn (Tax Rate POS Download) .....                 | 737        |
| txrtupld (Tax Rate Upload).....                        | 738        |
| <b>35 Tickets and Labels Batch.....</b>                | <b>741</b> |
| Overview.....                                          | 741        |
| Batch Design Summary .....                             | 741        |
| tcktdnld (Ticket Download).....                        | 741        |
| <b>36 Transfers, Allocation, and RTV Batch.....</b>    | <b>746</b> |
| Overview.....                                          | 746        |
| Transfers .....                                        | 746        |
| Returns to Vendor (RTVs).....                          | 747        |
| Batch Design Summary .....                             | 747        |
| distropcpub (Distro Price Change Publish).....         | 748        |
| docclose (Document Close) .....                        | 749        |
| mrt (Mass Return Transfer) .....                       | 750        |
| mrtprg (Mass Return Transfer Purge).....               | 752        |
| mrtrtv (Mass Return to Vendor Creation).....           | 755        |
| mrtupd (Mass Return Transfer Update) .....             | 756        |
| rtvprg (Return to Vendor Purge).....                   | 757        |
| tsfprg (Transfer Purge).....                           | 758        |
| <b>37 Value Added Tax (VAT) Maintenance Batch.....</b> | <b>761</b> |
| Overview.....                                          | 761        |
| System Level VAT .....                                 | 761        |
| System Class Level VAT.....                            | 761        |
| Department VAT .....                                   | 761        |
| Class VAT .....                                        | 762        |
| Store VAT Indicator .....                              | 762        |
| Send VAT Rate to POS .....                             | 762        |
| Special Note: Retail Method Stock Ledger and VAT.....  | 762        |
| Batch Design Summary .....                             | 762        |
| vatdlxpl (VAT Download Explode).....                   | 763        |

---

## Preface

Oracle Retail Operations Guides are designed so that you can view and understand the application's 'behind-the-scenes' processing. This volume of the Oracle Retail Merchandising System (RMS) Operations Guide includes the following information:

- Functional overviews related to batch processing
- Batch designs

## Audience

This operations guide is designed for System Analysts and Database Administrators who are looking for technical descriptions of data processes by functional area.

## Related Documents

You can find more information about this product in these resources:

- Oracle Retail Merchandising System Operations Guide – Volume 2
- Oracle Retail Merchandising System Operations Guide – Volume 3
- Oracle Retail Merchandising System Installation Guide
- Oracle Retail Merchandising System Release Notes
- Oracle Retail Merchandising System Online Help
- Oracle Retail Merchandising System User Guide
- Oracle Retail Merchandising System Data Model
- Oracle Retail Merchandising System Batch Schedule
- Oracle Retail Integration Bus (RIB) documentation
- Oracle Retail Service Layer (RSL) documentation
- Oracle Retail Extract, Transform and Load (RETL) documentation
- Other Oracle Retail product documentation

## Customer Support

- <https://metalink.oracle.com>

When contacting Customer Support, please provide:

- Product version and program/module name.
- Functional and technical description of the problem (include business impact).
- Detailed step-by-step instructions to recreate.
- Exact error message received.
- Screen shots of each step you take.





---

## Introduction

Welcome to the Oracle Retail Merchandising Operations Guide. The guide is designed to inform you about the ‘backend’ of RMS: data inputs, processes, and outputs. As a member of the Oracle Retail 12 family, RMS provides the many benefits of enterprise application integration (EAI).

A primary benefit of EAI is the near real-time view of data that results from message-based processes between RMS and other products on the Oracle Retail Integration Bus (RIB). RIB integration allows RMS to overcome time lags to data updates. As a result, RMS is less dependent upon the batch window.

### What’s in the guide

The major components of the Operations Guide include the three volumes described below.

#### **Volume 1 – Batch Overviews and Designs**

Batch overviews tie a functional area description to the batch processes illustrated in the designs. The overviews allow the reader to quickly determine how a business function works ‘behind the scenes.’

Batch designs describe how, on a technical level, an individual batch module works and the database tables that it affects. In addition, batch designs contain file layout information that is associated with the batch process. Note that the Oracle Retail Sales Audit (ReSA) Interface File Layout [rtlog] is located in the design, ‘saimptlog (Sales Audit Import)’.

Batch designs can be referenced by name through the table of contents of this volume.

#### **Volume 2 – Message Publication and Subscription Designs**

Oracle Retail Integration Bus (RIB) RMS functional overviews are incorporated into the publication and subscription designs. Therefore, the retailer can extract the business rationale behind each publication or subscription as well as the technical details that describe, on a technical level, how RMS publishes messages to the RIB or how RMS subscribes to messages from the RIB. A chapter in this volume also addresses how RMS utilizes the Oracle Retail Service Layer (RSL).

#### **A Note about ‘External’ Subscription RIB APIs**

Subscription APIs that are designated as ‘External’ are designed to be interfaces for external systems that maintain the applicable data. In other words, RMS is not the ‘system of record’ for maintaining the data. Instead, RMS subscribes to consume the data when it is published so that the corresponding data in RMS can be kept in sync with the external system that maintains the data.

## **Volume 3 – Backend Configuration and Operations**

This volume describes the important features that necessary to run the Pro\*C programs and the RETL programs associated with RMS. Additional RMS configuration and operations information is also included in this volume. Topics include:

- Pro\*C Restart and Recovery
- Pro\*C Multi-Threading
- Pro\*C Array Processing
- Pro\*C Input and Output Formats
- RETL Program Overview for RMS/ReSA Extractions
- RMS Internationalization and Localization

## **RMS Modules**

For RMS retailers who purchase additional modules, the guide includes descriptions of the batch programs related to the following:

- Oracle Retail Sales Audit™ (ReSA)
- Oracle Retail Trade Management™ (RTM)

## **The SYSTEMS\_OPTIONS Table**

Refer to the data model for information on the SYSTEM\_OPTIONS table. The SYSTEM\_OPTIONS table contains significant retailer-defined parameters. This table is populated during installation of the system and must be maintained by the database administrator.

---

## Audit Trail Batch

### Overview

The audit trail batch component performs two major functions:

- Activates or deactivates the audit trail functionality for the appropriate tables.
- Purges old information from the audit tables according to a predetermined schedule.

### Batch Designs Summary

The following batch designs are included in this functional area:

- AUDITPRG.PC (Audit Purge Process)
- AUDITSYS.PC (Audit Logic Information Edits)

### auditprg (Audit Purge Process)

#### Functional Area

Audit Trail

#### Module Affected

AUDITPRG.PC

#### Design Overview

The audit purge process truncates auditing tables based on the purge frequency code specified on the audit table (AUDIT\_TBL).

#### Scheduling Constraints

| Schedule Information      | Description                                                                             |
|---------------------------|-----------------------------------------------------------------------------------------|
| Processing Cycle          | AD-HOC (daily)                                                                          |
| Scheduling Considerations | This job is scheduled depending on the type of information to be audited in the system. |
| Pre-Processing            | N/A                                                                                     |
| Post-Processing           | N/A                                                                                     |
| Threading Scheme          | N/A                                                                                     |

#### Restart/Recovery

N/A

#### Locking Strategy

N/A

### Security Considerations

This program requires special permissions. It must be run by a DBA or an Oracle user that is granted the following privileges.

- 'drop any table' AND
- 'drop any trigger'

### Performance Considerations

N/A

### Key Tables Affected

| Table     | Select | Insert | Update | Delete |
|-----------|--------|--------|--------|--------|
| AUDIT_TBL | Yes    | No     | No     | No     |
| CALENDAR  | Yes    | No     | No     | No     |
| PERIOD    | Yes    | No     | No     | No     |

### I/O Specification

N/A

## auditsys (Audit Logic Information Edits)

### Functional Area

Audit Trails

### Module Affected

AUDITSYS.PC

### Design Overview

This program adds audit logic for a given table. A table that has an audit request raised against it has an audit table created to hold all audit details for any inserts, updates or deletes of data. This audit table (named [master\_table]\_AU) holds the key values of the master table and the username and date of the audited transaction. Additional fields to be audited may be added or removed at the user's request prior to running auditsys. Once the audit table is created, the AUDITPRG program must be run to remove the audit table, and then added to the RMS audit trail again for any changes to the columns being tracked. The audit module creates a database trigger to be applied to the master table.

The audit table and the database trigger are automatically promoted to the database. A user that is granted the following special privileges, or a user that has DBA privileges must execute this program:

- 'create any table'
- AND
- 'create any trigger'

## Scheduling Constraints

| Schedule Information      | Description                                                         |
|---------------------------|---------------------------------------------------------------------|
| Processing Cycle          | Phase Ad Hoc (daily)                                                |
| Scheduling Considerations | Needs to be scheduled only when new audit information is requested. |
| Pre-Processing            | N/A                                                                 |
| Post-Processing           | N/A                                                                 |
| Threading Scheme          | N/A (single threaded)                                               |

## Restart/Recovery

Used for logging purposes only. Because this program is applying ddl, that ddl cannot be rolled back. Tables and/or triggers should be dropped manually, or you can use the AUDITPRG.PC program to remove them.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS     | Yes    | No     | No     | No     |
| AUDIT_TBL          | Yes    | No     | No     | No     |
| ALL_TABLES         | Yes    | No     | No     | No     |
| ALL_INDEXES        | Yes    | No     | No     | No     |
| ALL_TAB_PARTITIONS | Yes    | No     | No     | No     |
| ALL_TAB_COLUMNS    | Yes    | No     | No     | No     |
| ALL_IND_COLUMNS    | Yes    | No     | No     | No     |
| AUDIT_FLD          | Yes    | No     | No     | No     |

## I/O Specification

N/A



## Overview

The system date is the current date used by RMS. You can set this date to correspond to the actual calendar date, or to a different date, as your system requires.

## Batch Design Summary

The following batch design is included in this functional area:

- DTESYS.PC (Increment Set System Date)

## dtesys (Increment Set System Date)

### Functional Area

Calendar batch

### Module Affected

DTESYS.PC

### Design Overview

This batch program updates the PERIOD table for various dates required in RMS such as vdate, end-of-month, and end-of-week dates. The flags listed below are also updated by this batch.

#### Daily:

If a date is passed-in as the second command line argument then the vdate is updated to the specified date. Otherwise, the vdate is incremented by one day.

#### Weekly:

When vdate = next\_eow\_date\_unit, the program increments the last\_eow\_date\_unit and next\_eow\_date\_unit columns on system\_variables. The last\_eow\_date\_unit is updated to the current next\_eow\_date\_unit and the next\_eow\_date\_unit is updated to the next end-of-week date (calculated).

#### Monthly:

When vdate = next\_eom\_date\_unit, the program updates the last\_eom\_date\_unit and next\_eom\_date\_unit columns on system\_variables. The last\_eom\_date\_unit is updated to the current next\_eom\_date\_unit and the next\_eom\_date\_unit is updated to the next end-of-month date (calculated).

**Scheduling Constraints**

| Schedule Information      | Description                                            |
|---------------------------|--------------------------------------------------------|
| Processing Cycle          | Date Set Phase – Daily                                 |
| Scheduling Considerations | This program should run at the end of the batch cycle. |
| Pre-Processing            | N/A                                                    |
| Post-Processing           | N/A                                                    |
| Threading Scheme          | N/A                                                    |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS   | Yes    | No     | No     | No     |
| PERIOD           | Yes    | No     | Yes    | No     |
| SYSTEM_VARIABLES | Yes    | No     | Yes    | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A



---

## Competitive Pricing Batch

### Overview

The RMS competitive pricing functionality extracts a competitor's price for an item and determines whether the price change is a candidate for export to Oracle Retail Price Management (RPM), where a price review is performed. Source data for the interface can be competitive prices manually entered directly into an RMS form (see the applicable competitor price entry window in RMS' online help) or a flat file uploaded by RMS' batch program CMPUPLD.PC. This document focuses on the process used by this batch program to upload, validate, and populate RMS tables.

---

**Note:** The flat file uploaded by CMPUPLD.PC can contain pricing data for a completed shopping list or data for a new list of items to be shopped. The module processes data for both features, as noted. However, the primary emphasis of this document is on the functionality that results in items that become candidates for export to Oracle Retail Price Management.

---

### Competitive Price Change Process

A competitor's regular or multi-unit price change for an item becomes a candidate for export from RMS to RPM. If the item price change is exported, RPM performs a price review that incorporates average sales data for the item at the affected store. RPM then transfers price change data to RMS' pricing tables. RMS completes the price change process by updating the store(s) through its point-of-sale (POS) download program POSDNLD.PC.

---

**Note:** For more information, see the chapter "Point-of-Sale (POS) Download Batch" in this volume of the RMS Operations Guide.

---

### Batch Design Summary

The following batch designs are included in this functional area:

- CMPPRG.PC (Competitive Pricing Purge)
- CMPUPLD.PC (Competitive Pricing Upload)

### cmpprg.pc (Competitive Pricing Purge)

#### Functional Area

Pricing

#### Module Affected

CMPPRG.PC

## Design Overview

This Competitive Pricing Purge program performs the desired deletions from the competitive price history table and the competitive shopping list table based purge criteria set on the SYSTEM\_OPTIONS table.

On the system options table, the comp\_price\_months field determines how many months competitive price history (comp\_price\_hist) is maintained before deletion. The comp\_list\_days field determines how long a requested shopping list (comp\_shop\_list) remains on the shopping list table if it is not complete by the requested shop date.

## Scheduling Constraints

| Schedule Information      | Description    |
|---------------------------|----------------|
| Processing Cycle          | AD-HOC (daily) |
| Scheduling Considerations | N/A            |
| Pre-Processing            | N/A            |
| Post-Processing           | N/A            |
| Threading Scheme          | N/A            |

## Restart/Recovery

N/A

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| PERIOD          | Yes    | No     | No     | No     |
| UNIT_OPTIONS    | Yes    | No     | No     | No     |
| COMP_PRICE_HIST | Yes    | No     | No     | Yes    |
| COMP_SHOP_LIST  | No     | No     | No     | Yes    |

## I/O Specification

N/A

## cmpupld (Competitive Pricing Upload)

### Functional Area

Competitive pricing

### Module Affected

CMPUPLD.PC

### Design Overview

CMPUPLD.PC is a Pro\*C program that runs as a module within RMS batch processing schedule. Its purpose is to upload and process competitor item prices from a competitive shopping list. The module accepts competitive shopping list data contained in a flat file (ASCII text) and formatted to match the prescribed retail input file format.

The item is validated against RMS' ITEM\_MASTER table. It cannot be above the transaction level. After all shopped items are validated, CMPUPLD.PC writes a row into the COMP\_SHOP\_LIST table.

The module functions in this way:

1. Verifies:
  - The competitor (validated against COMP\_STORE, COMPETITOR tables)
  - The competitor store (validated against COMP\_STORE table)
  - Shop date (validated value exists)
  - Shopper (validated against COMP\_SHOPPER table)
  - Item (validated against ITEM\_MASTER table)
2. Verifies the competitive retail price, the recorded date, and the competitive retail type either all exist or all do not exist.
3. Checks if the retail type is a regular price ('R'), promotional price ('P'), or clearance price ('C'). Only regular price changes become candidates for export to RPM. A competitor's promotional price or clearance price is not considered.

The item is validated against RMS' ITEM\_MASTER table. It cannot be above the transaction level and must contain a valid item code (item\_number\_type column in the ITEM\_MASTER table) of the type UPCT. After all shopped items are validated, CMPUPLD.PC writes a row into the COMP\_SHOP\_LIST table.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                            |
|---------------------------|------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Ad Hoc Interfaces                                                                                                      |
| Scheduling Considerations | This upload program should be scheduled to run before any of the Oracle Retail Pricing Management (RPM) batch modules. |
| Pre-Processing            | N/A                                                                                                                    |
| Post-Processing           | N/A                                                                                                                    |
| Threading Scheme          | The number of threads are based on the number of input files.                                                          |

**Restart/Recovery**

Since this is a file based upload, file based restart/recovery logic is applied. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 10000 records (subject to change based on experimentation).

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| COMP_SHOP_LIST | Yes    | Yes    | No     | No     |
| ITEM_MASTER    | Yes    | No     | No     | No     |
| CODE_DETAIL    | Yes    | No     | No     | No     |

**I/O Specification****Input File Layout**

| Record Name | Field Name                  | Field Type  | Default Value | Description                                                                                     |
|-------------|-----------------------------|-------------|---------------|-------------------------------------------------------------------------------------------------|
| File Header | File Type Record Descriptor | CHAR (5)    | FHEAD         | Value that identifies the record type.                                                          |
|             | File Line Identifier        | NUMBER (10) | 000000001     | Sequential file line number                                                                     |
|             | File Type Definition        | CHAR(4)     | CMPU          | Value that identifies the file as “Competitive Pricing Upload”.                                 |
|             | File Create Date            | CHAR (14)   |               | Date when the file was written by external system. It should be in the YYYYMMDDHH24MISS format. |
| File Detail | File Type Record Descriptor | CHAR(5)     | FDETL         | Value that identifies the record type.                                                          |
|             | File Line Identifier        | NUMBER(10)  |               | Sequential file line number                                                                     |

| Record Name | Field Name               | Field Type    | Default Value | Description                                                                                                                       |
|-------------|--------------------------|---------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------|
|             | Shopper ID               | NUMBER(4)     |               | Numeric value that uniquely identifies the shopper to which the competitive shopping list is assigned.                            |
|             | Shop Date                | CHAR(14)      |               | Date when the competitive shopping is performed. It should be in the YYYYMMDDHH24MISS format.                                     |
|             | Item                     | CHAR (25)     |               | Alphanumeric value that uniquely identifies the transaction level or below transaction level item that was competitively shopped. |
|             | Competitor ID            | NUMBER(10)    |               | Numeric value that uniquely identifies a competitor.                                                                              |
|             | Competitor Store ID      | NUMBER(10)    |               | Numeric value that uniquely identifies a competitor's store.                                                                      |
|             | Recorded Date            | CHAR (14)     |               | Date when the item's retail price is recorded at the competitor's store. It should be in the YYYYMMDD24MISS format.               |
|             | Competitive Retail Price | NUMBER(20, 4) |               | Numeric value that represents the retail price at the competitor's store. There should be four (4) implied decimal places.        |
|             | Competitive Retail Type  | CHAR(6)       | R, P, C       | Value that represents the retail type ('R' is for regular; 'P', promotional; and 'C', clearance) that is recorded.                |

| Record Name  | Field Name                  | Field Type    | Default Value | Description                                                                                                                                                                                                                                                             |
|--------------|-----------------------------|---------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              | Promotion Start Date        | CHAR (14)     | NULL          | Effective start date of the competitor's price. It should be in the YYYYMMDDHH24MISS format.                                                                                                                                                                            |
|              | Promotion End Date          | CHAR (14)     | NULL          | Effective end date of the competitor's price. It should be in the YYYYMMDDHH24MISS format.                                                                                                                                                                              |
|              | Offer Type Code             | CHAR(6)       | NULL          | Alphanumeric value that corresponds to a valid offer type (for example, Coupon, Bonus Card, Pre-priced). Valid values are defined on CODE_DETAILS table with code_type 'OFTP'.                                                                                          |
|              | Multi-Units                 | NUMBER(12, 4) |               | Numeric value that represents the number of units (for example, 2 for, 3 for) selling for a given amount (Multi-unit retail) if a multiple pricing method was in place for the item when it was competitively shopped. There should be four (4) implied decimal places. |
|              | Multi-Units Retail          | NUMBER(20, 4) |               | Numeric value that represents the amount of all the units selling if a multiple pricing method was in place for the item when it was competitively shopped. There should be four (4) implied decimal places.                                                            |
| File Trailer | File Type Record Descriptor | CHAR(5)       | FTAIL         | Value that identifies the record type.                                                                                                                                                                                                                                  |
|              | File Line Identifier        | NUMBER (10)   |               | Sequential file line number                                                                                                                                                                                                                                             |
|              | File Record Counter         | NUMBER (10)   |               | Numeric value that represents the number of FDETL records in the file.                                                                                                                                                                                                  |

## Contracts Batch

### Overview

Contract batch modules create purchase orders from contracts and purge obsolete contracts. A purchase order created from a contract has two primary differences from all other purchase orders in RMS. First, the only impact upon the order is the contract. Bracket costing and deals are not involved in a contract purchase order. Second, the cost of items on the order is predefined in the contract and is held at the item-supplier level.

There are four types of supplier contracts in RMS: A, B, C, and D.

- **Type A (Plan/Availability):** The contract contains a plan of manufacturing quantity by ready date. Supplier availability is matched to the ready date. Orders are raised against the plan as suggested by replenishment requirements, provided there is sufficient supplier availability. The user can also raise manual orders.
- **Type B (Plan/No Availability):** The contract contains a plan of manufacturing quantity by ready date and dispatch-to location or locations. There are one or more ready dates, which is the date that the items are due at the dispatch-to location. Supplier availability is not required. Orders are raised automatically from the contract based on ready dates.
- **Type C (No Plan/No Availability):** The contract is an open contract with no production schedule and no supplier availability declared. The contract lists the items that are used to satisfy a total commitment cost. Orders are raised against the contract based on replenishment requirements. The retailer can also raise manual orders.
- **Type D (No Plan/Availability):** The contract is an open contract with no production schedule. The supplier declares availability as stock is ready. The contract lists the items that are used to satisfy a total commitment cost. Orders are raised against the contract, based on replenishment requirements and supplier availability. The retailer can raise manual orders.

### Batch Design Summary

The following batch designs are included in this functional area:

---

**Note:** The batch program, EDIDLCON.PC (EDI Contract), has a functional connection to this chapter. For the design, please see the chapter, “Electronic Data Interchange (EDI) Batch,” in this volume of the Operations Guide.

**Note:** The batch program, CNTRPRSS.PC (Contract Replenishment), has a functional connection to this chapter. For the design, please see the chapter, “Replenishment Batch” in this volume of the Operations Guide.

---

- CNTRMAIN.PC (Contract Maintenance and Purging)
- CNTRORDB.PC (Contract Replenishment–Type B Contracts)

## cntrmain (Contract Maintenance and Purging)

### Functional Area

Contracts

### Module Affected

CNTRMAIN.PC

### Design Overview

This module purges contracts that have remained in cancelled, worksheet, submitted or complete status for a user-defined number of months and if there are no existing orders against it. Additionally, 'Approved' closed (types A and B) contracts are set to 'Complete' status while those that are of type C and D are set to 'Review' status automatically.

### Scheduling Constraints

| Schedule Information      | Description                                         |
|---------------------------|-----------------------------------------------------|
| Processing Cycle          | Phase 0, Daily                                      |
| Scheduling Considerations | Needs to be scheduled before replenishment modules. |
| Pre-Processing            | N/A                                                 |
| Post-Processing           | N/A                                                 |
| Threading Scheme          | N/A                                                 |

### Restart/Recovery

This batch program has two processing functions, one for purging and another for updating contracts. The purge function (delete\_contracts) deletes and commits records via arrays whose size is defined in commit max counter while the update function (reset\_inactive) updates records in bulk based on the update criteria. The program as a whole is inherently restartable.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A



**Key Tables Affected**

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| UNIT_OPTIONS    | Yes    | No     | No     | No     |
| CONTRACT_HEADER | Yes    | No     | Yes    | Yes    |
| CONTRACT_DETAIL | No     | No     | No     | Yes    |
| CONTRACT_COST   | No     | No     | No     | Yes    |
| ORDHEAD         | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**cntrordb (Contract Replenishment Type 'B')****Functional Area**

Contracts

**Module Affected**

CNTRORDB.PC

**Design Overview**

This module automatically creates replenishment orders for all approved type 'B' contract items where orderable indicator is set to be 'Y' (Yes).

Contract, item, and location data is selected from the CONTRACT\_HEADER and CONTRACT\_DETAILS tables where production dates are ready to be met. The module writes an order for each contract to include all items and locations on the contract. The module runs if the contract replenishment indicator (contract\_replenish\_ind column in the SYSTEM\_OPTIONS table) is set to 'Y' (Yes) and type B contracts are used.

**Scheduling Constraints**

| Schedule Information      | Description                                                                     |
|---------------------------|---------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                                 |
| Scheduling Considerations | Must be run after REPLADJ                                                       |
| Pre-Processing            | N/A                                                                             |
| Post-Processing           | Prepost cntrordb post - update of system_variables.last_cont_order_date = vdate |
| Threading Scheme          | This module is threaded by contract                                             |

**Restart/Recovery**

The logical unit of work is contract number. Records are committed to the database when number of records processed reaches commit\_max\_counter maintained in RESTART\_CONTROL table.

**Locking Strategy**

ORDLOC\_EXP records are locked during processing.

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| UNIT_OPTIONS          | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES      | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS        | Yes    | No     | No     | No     |
| PERIOD                | Yes    | No     | No     | No     |
| CONTRACT_HEADER       | Yes    | No     | No     | No     |
| CONTRACT_DETAIL       | Yes    | No     | Yes    | No     |
| ORDHEAD               | Yes    | Yes    | Yes    | No     |
| ORDSKU                | Yes    | Yes    | Yes    | No     |
| ORDLOC                | Yes    | Yes    | Yes    | No     |
| ORDLOC_EXP            | Yes    | No     | Yes    | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | Yes    | No     | No     | No     |
| SUPS                  | Yes    | No     | No     | No     |
| WH                    | Yes    | No     | No     | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| ITEM_LOC              | Yes    | Yes    | No     | No     |
| PRICE_HIST            | No     | Yes    | No     | No     |
| ITEM_LOC_SOH          | No     | Yes    | No     | No     |
| SUP_INV_MGMT          | Yes    | No     | No     | No     |
| ORD_INV_MGMT          | No     | Yes    | No     | No     |
| ITEM_EXP_DETAIL       | No     | No     | Yes    | No     |
| ITEM HTS_ASSESS       | No     | No     | Yes    | No     |
| ORDSKU HTS_ASSESS     | No     | No     | Yes    | No     |

**I/O Specification**

N/A

## Cost Change Batch

### Overview

Cost values serve as a starting point in the creation of a purchase order. RMS uses the multi-channel concept where stores and warehouses can be 'virtual' as well as physical locations. If RMS is set up to run multi-channel (meaning the multi-channel indicator on the SYSTEM\_OPTIONS table is set to 'Y' [Yes]), only virtual locations hold stock. Physical warehouses, although not stockholding locations, do hold supplier item cost information that is shared across all virtual warehouses associated with the physical warehouse. This section describes how supplier cost changes are processed in RMS with a focus on the batch modules SCCEXT.PC and CCPRG.PC.

### Cost Change Process

Cost changes made through the front end of RMS impact these tables:

- COST\_SUSP\_SUP\_HEAD, always populated
- COST\_SUSP\_SUP\_DETAIL, populated if the cost at the country level is changed. Otherwise COST\_SUSP\_SUP\_DETAIL\_LOC is populated if cost is being maintained at individual locations. Bracket cost data is also stored on these two tables.

If cost changes are updated directly from the supplier, the batch module EDIUPCAT.PC indirectly populates the cost tables. EDIUPCAT.PC populates EDI\_COST\_CHG and EDI\_COST\_LOC. The RMS user can then accept EDI cost changes through the EDI cost change dialog. Accepted changes then populate the cost tables.

---

**Note:** The EDIUPCAT.PC batch module pertains only to retailers who use Electronic Data Interchange (EDI).

---

After updates to the cost tables occur, they are processed into the following tables by the SCCEXT.PC module:

- ITEM\_SUP\_COUNTRY for the country level cost change. The program distributes the cost to all locations on ITEM\_SUP\_COUNTRY\_LOC (for the current unit cost). Note that this table is always updated, regardless of the multi-channel indicator.
- ITEM\_SUPP\_COUNTRY\_BRACKET\_COST if the supplier is bracket costing.
- ITEM\_LOC\_SOH for locations .

### Multi-channel Supplier Cost Change Rules:

- Average cost is held on the ITEM\_LOC\_SOH table
- Cost changes are managed and stored at the physical warehouse level because the unit cost must remain consistent across all virtual warehouses within the same physical warehouse
- On the ITEM\_LOC\_SOH table, cost is held at the virtual level, to include physical stores
- A purchase order PO cannot be created for non-stockholding locations, such as physical warehouses, and non-stockholding stores (for example, Web stores and catalog stores)

- Each physical and virtual store has a default virtual warehouse
- Cost changes sent by a supplier and uploaded by the batch module EDIUPCAT.PC apply to the physical warehouse before quantities are apportioned to the virtual warehouses in SCCEXT.PC
- When cost changes are received from a supplier via EDI, two outcomes are possible for updating the system costs. If the item is in Worksheet or Submitted status, system costs are updated online when the cost change is accepted in the EDI dialog. If the item is in Accepted status, the cost change records are written to the cost change dialog. From there, when the cost change is approved, SCCEXT.PC processes these cost changes and updates system costs.

## Batch Design Summary

The following batch designs are included in this functional area:

---

**Note:** The batch program, EDIUPCAT.PC, has a functional connection to this chapter. For the design, please see the chapter, “Electronic Data Interchange (EDI) Batch,” in this volume of the Operations Guide.

---

- CCPRG.PC (Cost Event Purge)
- SCCEXT.PC (Supplier Cost Change Extract)

## ccprg (Cost Event Purge)

### Functional Area

Pricing

### Module Affected

CCPRG.PC

### Design Overview

This program is responsible for removing old cost changes from the system. Cost changes are removed from the system using the following criteria:

- The status of the cost change is Delete, Canceled, or Extracted
- The status of the price change is Rejected and the effective date of the cost change has met the requirement for the number of days that rejected cost changes are held.

The number of days that rejected price changes are held is determined by the system option, number\_days\_rejects\_held on the UNIT\_OPTIONS table.

**Scheduling Constraints**

| Schedule Information      | Description      |
|---------------------------|------------------|
| Processing Cycle          | AD-HOC (monthly) |
| Scheduling Considerations | N/A              |
| Pre-Processing            | N/A              |
| Post-Processing           | N/A              |
| Threading Scheme          | N/A              |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                    | Select | Insert | Update | Delete |
|--------------------------|--------|--------|--------|--------|
| PERIOD                   | Yes    | No     | No     | No     |
| COST_SUSP_SUP_HEAD       | Yes    | No     | No     | Yes    |
| COST_SUSP_SUP_DETAIL     | Yes    | No     | No     | Yes    |
| COST_SUSP_SUP_DETAIL_LOC | Yes    | No     | No     | Yes    |
| UNIT_OPTIONS             | Yes    | No     | No     | No     |

**I/O Specification**

N/A

## sccect (Supplier Cost Change Extract)

### Functional Area

Cost change

### Module Affected

SCCEXT.PC

### Design Overview

The Supplier Cost Change Extract module selects supplier cost change records, which are set to go into effect the next day and updates the RMS item/supplier table with the new cost. The item/location tables are also updated with the new cost if the cost change is a base cost change (supplier is the primary supplier for the item).

### Scheduling Constraints

| Schedule Information      | Description                                                                        |
|---------------------------|------------------------------------------------------------------------------------|
| Processing Cycle          | AD HOC                                                                             |
| Scheduling Considerations | N/A                                                                                |
| Pre-Processing            | N/A                                                                                |
| Post-Processing           | prepost sccect post – sets the status of processed cost changes to extracted ('E') |
| Threading Scheme          | Threaded by cost change.                                                           |

### Restart/Recovery

The logical unit of work for the program is a cost change. The program is also restartable from the last successfully processed cost change record.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| COST_SUSP_SUP_HEAD   | Yes    | No     | No     | No     |
| DEAL_CALC_QUEUE_TEMP | Yes    | No     | No     | No     |
| DEAL_CALC_QUEUE      | Yes    | Yes    | Yes    | No     |
| PERIOD               | Yes    | No     | No     | No     |

| Table                          | Select | Insert | Update | Delete |
|--------------------------------|--------|--------|--------|--------|
| ITEM_SUPP_COUNTRY_LOC          | Yes    | No     | Yes    | No     |
| COST_SUSP_SUP_DETAIL           | Yes    | No     | No     | No     |
| DEAL_SKU_TEMP                  | No     | Yes    | No     | No     |
| PRICE_HIST                     | No     | Yes    | No     | No     |
| ITEM_SUPPLIER                  | Yes    | No     | Yes    | No     |
| SUPS                           | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS                 | Yes    | No     | No     | No     |
| COST_SUSP_SUP_DETAIL_LOC       | Tes    | No     | No     | No     |
| ORDLOC                         | Yes    | No     | Yes    | No     |
| ORDHEAD                        | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY              | Yes    | No     | Yes    | No     |
| ITEM_SUPP_COUNTRY_BRACKET_COST | Yes    | Yes    | Yes    | No     |
| ITEM_MASTER                    | Yes    | No     | No     | No     |
| PACKITEM                       | Yes    | No     | No     | No     |

### I/O Specification

N/A

---

## Daily Purge Batch

### Overview

Daily purge processing spans multiple functional areas. The batch program is a general system maintenance program.

### Batch Design Summary

The following batch design is included in this functional area:

- DLYPRG.PC (Daily Purge)

### dlyprg (Daily Purge)

#### Functional Area

Maintenance Daily purge processing spans multiple functional areas. The batch program is a general system maintenance program.

#### Module Affected

DLYPRG.PC

#### Design Overview

The purpose of this program is to delete all of the records in the system marked for deletion (by having a record on the DAILY\_PURGE table) during the day. Before deleting the records, all relations are checked to ensure that the record can be deleted. For example, if an item is marked for deletion, this program checks that the item was not put on order later in the day. If relations are found to exist, a record is written to an error table. Records on this table are used to generate a report itemizing any problems found when running this program.

#### Scheduling Constraints

| Schedule Information      | Description                                                       |
|---------------------------|-------------------------------------------------------------------|
| Processing Cycle          | Phase 0, daily                                                    |
| Scheduling Considerations | This program must run first to avoid processing deleted entities. |
| Pre-Processing            | N/A                                                               |
| Post-Processing           | N/A                                                               |
| Threading Scheme          | N/A                                                               |

#### Restart/Recovery

This program has inherent restart ability. Records that are successfully purged are deleted from the DAILY\_PURGE table. This ensures that if the program is restarted, it does not attempt to delete records that were previously processed.



## Locking Strategy

This module locks all the tables from which it is purging.

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| DAILY_PURGE           | Yes    | No     | No     | Yes    |
| DAILY_PURGE_ERROR_LOG | Yes    | Yes    | No     | Yes    |
| LOC_LIST_DETAIL       | No     | No     | No     | Yes    |
| MONTH_DATA_BUDGET     | Yes    | No     | No     | Yes    |
| HALF_DATA_BUDGET      | Yes    | No     | No     | Yes    |
| VAT_DEPS              | Yes    | No     | No     | Yes    |
| SKULIST_CRITERIA      | Yes    | No     | No     | Yes    |
| STORE_DEPT_AREA       | Yes    | No     | No     | Yes    |
| DOMAIN_DEPT           | Yes    | No     | No     | Yes    |
| FORECAST_REBUILD      | Yes    | No     | No     | Yes    |
| SUP_DATA              | Yes    | No     | No     | Yes    |
| DEPT_SALES_HIST       | Yes    | No     | No     | Yes    |
| DEPT_SALES_FORECAST   | Yes    | No     | No     | Yes    |
| DEAL_ITEMLOC          | Yes    | No     | No     | Yes    |
| DEPS                  | Yes    | No     | No     | Yes    |
| STOCK_LEDGER_INSERTS  | Yes    | No     | No     | Yes    |
| STAKE_SCHEDULE        | Yes    | No     | No     | Yes    |
| PRODUCT_TAX_CODE      | Yes    | No     | No     | Yes    |
| DEPT_CHRG_DETAIL      | Yes    | No     | No     | Yes    |
| WH_DEPT               | Yes    | No     | No     | Yes    |
| DEPT_CHRG_HEAD        | Yes    | No     | No     | Yes    |
| SUP_BRACKET_COST      | Yes    | No     | No     | Yes    |
| SUP_REPL_DAY          | Yes    | No     | No     | Yes    |
| SUP_INV_MGMT          | Yes    | No     | No     | Yes    |
| FILTER_GROUP_MERCH    | Yes    | No     | No     | Yes    |

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| IB_RESULTS              | Yes    | No     | No     | Yes    |
| WEEK_DATA               | Yes    | No     | No     | Yes    |
| DAILY_DATA              | Yes    | No     | No     | Yes    |
| MONTH_DATA              | Yes    | No     | No     | Yes    |
| TRAN_DATA_HISTORY       | Yes    | No     | No     | Yes    |
| HALF_DATA               | Yes    | No     | No     | Yes    |
| PARTNER                 | Yes    | No     | No     | Yes    |
| SHIPMENT                | Yes    | No     | No     | Yes    |
| COST_ZONE_GROUP_LOC     | Yes    | No     | No     | Yes    |
| COST_ZONE               | Yes    | No     | No     | Yes    |
| COST_ZONE_GROUP         | Yes    | No     | No     | Yes    |
| UDA_ITEM_DEFAULTS       | Yes    | No     | No     | Yes    |
| DOMAIN_CLASS            | Yes    | No     | No     | Yes    |
| CLASS_SALES_HIST        | Yes    | No     | No     | Yes    |
| CLASS_SALES_FORECAST    | Yes    | No     | No     | Yes    |
| CLASS                   | Yes    | No     | No     | Yes    |
| DOMAIN_SUBCLASS         | Yes    | No     | No     | Yes    |
| OTB_FWD_LIMIT           | Yes    | No     | No     | Yes    |
| OTB                     | Yes    | No     | No     | Yes    |
| DIFF_RATIO_DETAIL       | Yes    | No     | No     | Yes    |
| DIFF_RATIO_HEAD         | Yes    | No     | No     | Yes    |
| SUBCLASS_SALES_HIST     | Yes    | No     | No     | Yes    |
| SUBCLASS_SALES_FORECAST | Yes    | No     | No     | Yes    |
| EDI_NEW_ITEM            | Yes    | No     | No     | Yes    |
| SUBCLASS                | Yes    | No     | No     | Yes    |
| MERCH_HIER_DEFAULT      | Yes    | No     | No     | Yes    |
| WH                      | Yes    | No     | No     | Yes    |
| WH_ATTRIBUTES           | Yes    | No     | No     | Yes    |
| WH_ADD                  | Yes    | No     | No     | Yes    |
| STORE_SHIP_DATE         | Yes    | No     | No     | Yes    |
| LOC_TRAITS_MATRIX       | Yes    | No     | No     | Yes    |
| SEC_USER_ZONE_MATRIX    | Yes    | No     | No     | Yes    |

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| PRICE_ZONE_GROUP_LOC    | Yes    | No     | No     | Yes    |
| COST_ZONE_GROUP_LOC     | Yes    | No     | No     | Yes    |
| ITEM_EXP_DETAIL         | Yes    | No     | No     | Yes    |
| ITEM_EXP_HEAD           | Yes    | No     | No     | Yes    |
| EXP_PROF_DETAIL         | Yes    | No     | No     | Yes    |
| EXP_PROF_HEAD           | Yes    | No     | No     | Yes    |
| STORE_ATTRIBUTES        | Yes    | No     | No     | Yes    |
| STORE_DEPT_AREA         | Yes    | No     | No     | Yes    |
| STORE_GRADE_STORE       | Yes    | No     | No     | Yes    |
| DAILY_SALES_DISCOUNT    | Yes    | No     | No     | Yes    |
| LOAD_ERR                | Yes    | No     | No     | Yes    |
| STORE                   | Yes    | No     | No     | Yes    |
| EDI_SALES_DAILY         | Yes    | No     | No     | Yes    |
| COMP_STORE_LINK         | Yes    | No     | No     | Yes    |
| REPL_RESULTS            | Yes    | No     | No     | Yes    |
| SEC_GROUP_LOC_MATRIX    | Yes    | No     | No     | Yes    |
| LOC_CLSF_HEAD           | Yes    | No     | No     | Yes    |
| LOC_CLSF_DETAIL         | Yes    | No     | No     | Yes    |
| SOURCE_DLVRY_SCHED      | Yes    | No     | No     | Yes    |
| SOURCE_DLVRY_SCHED_DAYS | Yes    | No     | No     | Yes    |
| SOURCE_DLVRY_SCHED_EXC  | Yes    | No     | No     | Yes    |
| COMPANY_CLOSED_EXCEP    | Yes    | No     | No     | Yes    |
| LOCATION_CLOSED         | Yes    | No     | No     | Yes    |
| GENCODE_STORE           | Yes    | No     | No     | Yes    |
| POS_STORE               | Yes    | No     | No     | Yes    |
| SUB_ITEMS_DETAIL        | Yes    | No     | No     | Yes    |
| SUB_ITEMS_HEAD          | Yes    | No     | No     | Yes    |
| STORE_HIERARCHY         | Yes    | No     | No     | Yes    |
| ADDR                    | Yes    | No     | No     | Yes    |
| TIF_EXPLODE             | Yes    | No     | No     | Yes    |
| WALK_THROUGH_STORE      | Yes    | No     | No     | Yes    |
| SKULIST_DETAIL          | Yes    | No     | No     | Yes    |

| Table                    | Select | Insert | Update | Delete |
|--------------------------|--------|--------|--------|--------|
| INV_STATUS_QTY           | Yes    | No     | No     | Yes    |
| REPL_ATTR_UPDATE_EXCLUDE | Yes    | No     | No     | Yes    |
| REPL_ATTR_UPDATE_LOC     | Yes    | No     | No     | Yes    |
| REPL_ATTR_UPDATE_HEAD    | Yes    | No     | No     | Yes    |
| MASTER_REPL_ATTR         | Yes    | No     | No     | Yes    |
| REPL_ATTR_UPDATE_ITEM    | Yes    | No     | No     | Yes    |
| REPL_DAY                 | Yes    | No     | No     | Yes    |
| REPL_ITEM_LOC            | Yes    | No     | No     | Yes    |
| REPL_ITEM_LOC_UPDATES    | Yes    | Yes    | No     | Yes    |
| COST_SUSP_SUP_DETAIL_LOC | Yes    | No     | No     | Yes    |
| COST_SUSP_SUP_DETAIL     | Yes    | No     | No     | Yes    |
| ITEM HTS_ASSESS          | Yes    | No     | No     | Yes    |
| ITEM HTS                 | Yes    | No     | No     | Yes    |
| REQ_DOC                  | Yes    | No     | No     | Yes    |
| ITEM_IMPORT_ATTR         | Yes    | No     | No     | Yes    |
| TIMELINE                 | Yes    | No     | No     | Yes    |
| COND_TARIFF_TREATMENT    | Yes    | No     | No     | Yes    |
| ITEM_IMAGE               | Yes    | No     | No     | Yes    |
| ITEM_SUPP_UOM            | Yes    | No     | No     | Yes    |
| DEAL_SKU_TEMP            | Yes    | No     | No     | Yes    |
| FUTURE_COST              | Yes    | No     | No     | Yes    |
| DEAL_DETAIL              | Yes    | No     | No     | Yes    |
| ITEM_SUPP_COUNTRY        | Yes    | No     | No     | Yes    |
| ITEM_SUPP_COUNTRY_DIM    | Yes    | No     | No     | Yes    |
| RECLASS_ITEM             | Yes    | No     | No     | Yes    |
| SUP_AVAIL                | Yes    | No     | No     | Yes    |
| ITEM_ATTRIBUTES          | Yes    | No     | No     | Yes    |
| ITEM_LOC                 | Yes    | No     | No     | Yes    |
| ITEM_LOC_SOH             | Yes    | No     | No     | Yes    |
| ITEM_SUPPLIER            | Yes    | No     | No     | Yes    |
| ITEM_MASTER              | Yes    | No     | No     | Yes    |
| PACK_TMPL_DETAIL         | Yes    | No     | No     | Yes    |

| Table                          | Select | Insert | Update | Delete |
|--------------------------------|--------|--------|--------|--------|
| SUPS_PACK_TMPL_DESC            | Yes    | No     | No     | Yes    |
| PACK_TMPL_HEAD                 | Yes    | No     | No     | Yes    |
| UDA_ITEM_LOV                   | Yes    | No     | No     | Yes    |
| UDA_ITEM_DATE                  | Yes    | No     | No     | Yes    |
| UDA_ITEM_FF                    | Yes    | No     | No     | Yes    |
| ITEM_SEASONS                   | Yes    | No     | No     | Yes    |
| ITEM_TICKET                    | Yes    | No     | No     | Yes    |
| COMP_SHOP_LIST                 | Yes    | No     | Yes    | Yes    |
| TICKET_REQUEST                 | Yes    | No     | No     | Yes    |
| PRODUCT_TAX_CODE               | Yes    | No     | No     | Yes    |
| PRICE_HIST                     | Yes    | Yes    | No     | Yes    |
| POS_MODS                       | Yes    | Yes    | No     | Yes    |
| ITEM_LOC_TRAITS                | Yes    | No     | No     | Yes    |
| PACKITEM_BREAKOUT              | Yes    | No     | No     | Yes    |
| PACKITEM                       | Yes    | No     | No     | Yes    |
| ITEM_SUPP_COUNTRY_BRACKET_COST | Yes    | No     | No     | Yes    |
| ITEM_SUP_COUNTRY_LOC           | Yes    | No     | No     | Yes    |
| EDI_COST_LOC                   | Yes    | No     | No     | Yes    |
| EDI_COST_CHG                   | Yes    | No     | No     | Yes    |
| POS_MERCH_CRITERIA             | Yes    | No     | No     | Yes    |
| ITEM_CHRG_HEAD                 | Yes    | No     | No     | Yes    |
| ITEM_CHRG_DETAIL               | Yes    | No     | No     | Yes    |
| RECLASS_COST_CHG_QUEUE         | Yes    | No     | No     | Yes    |
| ITEM_PUB_INFO                  | Yes    | No     | No     | Yes    |
| ITEM_MFQUEUE                   | Yes    | No     | No     | Yes    |
| ITEM_XFORM_HEAD                | Yes    | No     | No     | Yes    |
| ITEM_XFORM_DETAIL              | Yes    | No     | No     | Yes    |
| DEAL_ITEM_LOC_EXPLODE          | Yes    | No     | No     | Yes    |
| ITEM_APPROVAL_ERROR            | Yes    | No     | No     | Yes    |

### I/O Specification

N/A

## Deals Maintenance Batch

### Overview

Deals in RMS apply to two areas of processing:

1. Deals that are set up and approved in the system are used to calculate and add discounts to ordered item-location combination costs on purchase orders. Costs for ordered item-location combinations are held on the ORDLOC table in the unit\_cost column. The retailer has the choice of applying deals to orders online, or during the nightly batch run.
2. Approved deals are also used to calculate estimated future cost for item-supplier-origin-country-active date-location combinations. These estimated future costs are held on the FUTURE\_COST table and are used by the Investment Buy modules.

This overview describes the processing of a deal, which involves batch modules that work toward populating the FUTURE\_COST table, and the application of deals to purchase orders, which involves batch modules that calculate and save the discounted costs on the ORDLOC table.

### Deal Concepts

A deal can be an off-invoice allowance, a bill-back, or a rebate. The four levels of costing in a deal that are calculated and inserted into the FUTURE\_COST table are:

**Base Cost**—The starting cost of an item, prior to any processing, or the cost stored at the item-supplier-country (on the ITEM\_SUPP\_COUNTRY table) or item-supplier-country-location (on the ITEM\_SUPP\_COUNTRY\_LOC table) level.

**Net Cost**—The cost after any off-invoice deals are applied. This is the cost that is sent to the supplier on the purchase order, as well as the cost that is used for invoice matching.

**Net-Net Cost**—The cost of an item following the application of any discounts associated with the item. Discounts are billback discounts that are offered and apply specifically to a purchase order.

**Dead Net-Net Cost**—Calculated as the final cost of the item. Along with the off-invoice deals and discounts that are applied, the dead net-net cost also includes any rebates for the item. Rather than being specific to an order, rebates are applied to all orders created during a specified time period, and are billed back at the end of the time period. This column is also used as an estimate for costs that appears on a purchase order on a given future date.

Within RMS, the batch programs recalculate these item cost levels based on the approved deals and the start and close date of the approved deals. These costs are then stored at the item-supplier-country-active date-location level on the FUTURE\_COST table.

### Types of Deals

There are three types of deals: off invoice, fixed deals, and bill-backs.

**Fixed deals**—Fixed deals are a lump sum deal based on proof of performance. For example, if a retailer highlights a particular product in their weekly sales flyer (and sends that flyer to the product's manufacturer as proof), then the manufacturer gives the retailer a \$100 discount on their order.

**Off invoice versus bill-back**—Both off-invoice and bill-back deals are based on a threshold. If the retailer meets or exceeds the threshold, then the deal is applied. Off-invoice deals are applied at the time of order. Bill-backs are deals that are applied after an order, receipt, or sale takes place, and are performance-based.

## Rules that Apply to Deal Functionality

Here is a quick reference list of the rules that RMS enforces within its batch modules for deals:

- All deals must be at the location level because cost is held at the location level.
- Deals only apply to physical warehouse locations. All the virtual warehouses within a physical warehouse share the same item cost.
- All applicable deals are always applied. Bill Back deals are always applied against the PO or receipt cost or the retail price of the sale. They are not applied on top or after each other.
- Packs (complex or simple) are only applied to purchase order based deals. Packs are not exploded down to their components for deal application.
- Outstanding purchase order item shipments not yet received are automatically recalculated for applicable deals by default, unless the retailer indicates otherwise when setting up a deal.
- Bracket costs and scaling are always applied if applicable before deals are applied to the order.

## Deal Process for Off-invoice

A deal is applied to purchase orders through the following steps. The batch programs for each step in the process are in parentheses. Each step is described along with its respective program(s) in greater detail in the sections that follow.

1. Define deals through the (DEALUPLD.PC) batch module, or online in RMS.
  - A deal is defined by the items, locations, and the terms and discounts that the trading partner—that is, the supplier—offers. The batch module DEALUPLD.PC or an RMS form can be used in this step.
2. Process Deals by populating the DEAL\_SKU\_TEMP and FUTURE\_COST tables and also populating DEAL\_CALC\_QUEUE, which holds orders that may be affected by changed deals.
  - Processing a deal requires use of deal attributes held in RMS. These attributes determine the cost of an item at a location at a point in time. Three batch programs process a deal in this order:
    - DITINSRT.PC
    - PRECOSTCALC.PC
    - COSTCALC.PC.
3. Apply deals to orders (ORDDSCNT.PC).
  - When a deal is actually applied to a purchase order, the item cost is reduced. A retailer can apply a deal online. The ORDDSCNT.PC module applies a deal within the batch schedule.
4. Clean up deals (DEALCLS.PC) and (DEALPRG.PC).
  - Two batch modules close expired deals and purge dated deals from RMS tables.

## Deal Process for Bill-backs

Bill-back deals are applied as follows:

1. Define deals online in RMS (DEALUPLD.PC can only be used for off invoice deals).
  - A deal is defined by the items, locations, and the terms and discounts that the trading partner – that is, the supplier, manufacturer, wholesaler or distributor – offers. Use an RMS form in this step.
2. Process deals by populating the following tables:
  - DEAL\_SKU\_TEMP
  - FUTURE\_COST
  - DEAL\_CALC\_QUEUE (which holds orders that may be affected by changed deals)

Processing a deal requires use of deal attributes held in RMS. These attributes determine the cost of an item at a location at a point in time. Three batch programs process a deal in this order:

- DITINSRT.PC
  - PRECOSTCALC.PC
  - COSTCALC.PC.
3. DEALEX.PC extracts item/location information out to the DEAL\_ITEM\_LOC\_EXPLODE table. SALSTAGE.PC, a secondary process, posts relevant sales and receipt transactions to the DEAL\_PERF\_TRAN\_DATA table. DEALACT.PC uses both tables to aggregate on a daily basis the turnover for each deal component. DEALACT.PC also scans for approved POs.
  4. At the end of the month, DEALINC.PC runs, and calculates income for each item/loc record. It runs before SALMTH.PC runs, and before the end of month processing. DEALINC.PC updates the DEAL\_ACTUALS\_ITEM\_LOC table. The income that DEALINC.PC calculates is then rolled up by DEALFCT.PC to the DEAL\_ACTUALS\_FORECAST table for each deal component.
  5. To post records to the general ledger and the stock ledger, run PREPOST.PC dealday pre (monthly), DEALDAY.PC (monthly), and PREPOST.PC dealday post.
  6. VENDINVC.PC runs daily. It posts invoice information when the estimated next invoice date is reached and income is generated for a reporting period. In addition, it posts income for complex deals. It fills in the last invoice date with the end date of the last reporting period that is processed.
  7. VENDINVF.PC runs daily. It posts invoice information when the estimated next invoice date is reached and income is generated for a reporting period. In addition, it posts income for fixed deals. It fills in the last invoice date with the end date of the last reporting period that is processed.
  8. Clean up deals with DEALCLS.PC and DEALPRG.PC. These two batch modules close expired deals and purge dated deals from RMS tables.

## Deals Process

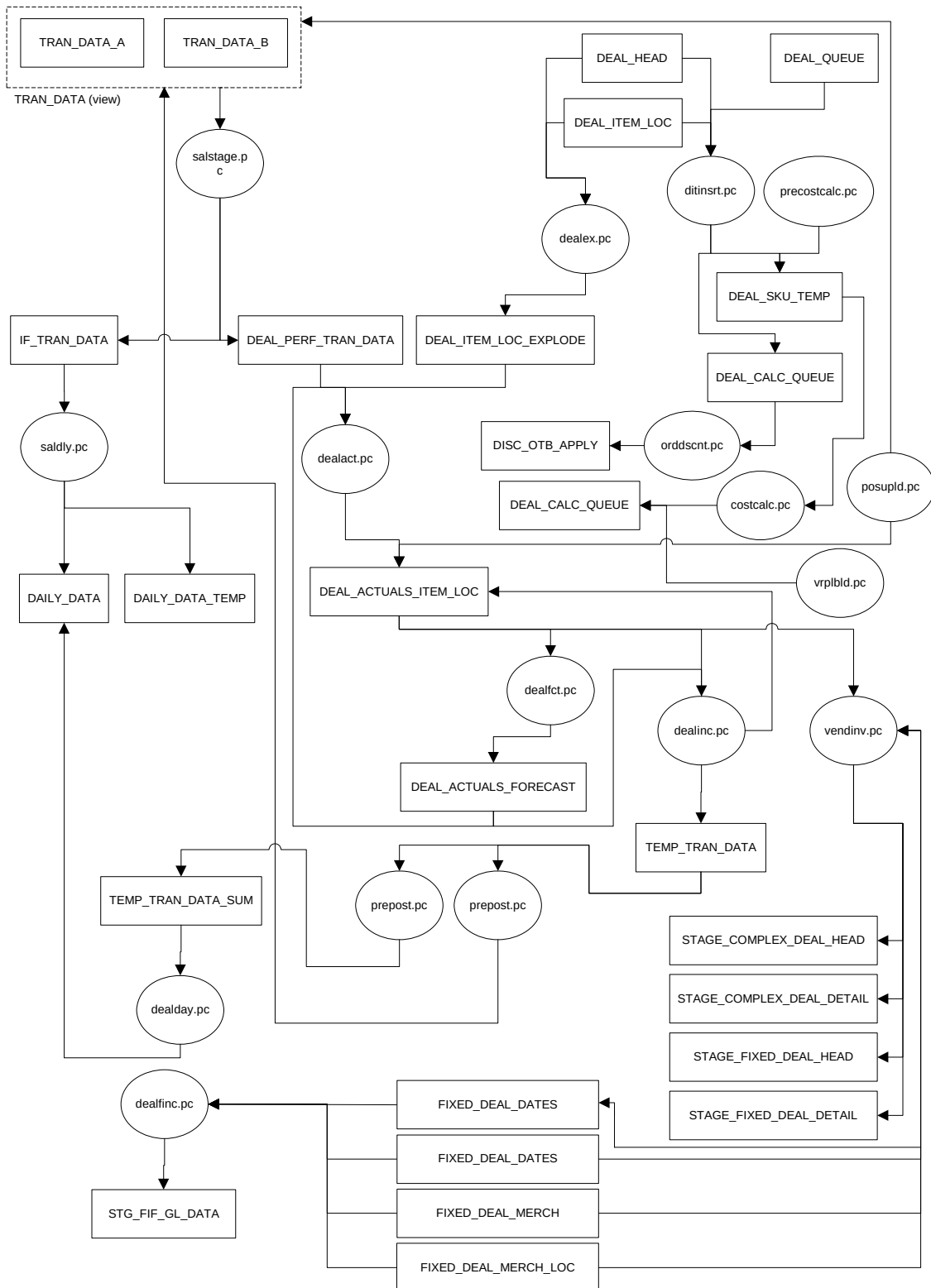
The diagram below depicts the deals process.

---

**Note:** The DEALUPLD.PC, DEALCLS.PC, and DEALPRG.PC batch programs are not included in the diagram for the sake of readability.

---





## Deals Processing

## Batch Design Summary

The following batch designs are included in this functional area:

---

**Note:** The batch program, POSUPLD.PC, has a functional connection to this chapter. For the design, please see the chapter, “Sales Posting Batch,” in this volume of the Operations Guide.

**Note:** The batch program, SALSTAGE.PC, has a functional connection to this chapter. For the design, please see the chapter, “Stock Ledger Batch,” in this volume of the Operations Guide.

---

- COSTCALC.PC (Deal Cost Calculation)
- DEALACT.PC (Deal Actuals)
- DEALCLS.PC (Deal Close)
- DEALDAY.PC (Deal Daily Data)
- DEALEX.PC (Deal Explode)
- DEALFCT.PC (Deal Forecast)
- DEALINC.PC (Deal Income)
- DEALPRG.PC (Deal Purge)
- DEALUPLD.PC (Deal Upload)
- DISCOTBAPPLY.PC (Discount OTB apply)
- DITINSRT.PC (Deal-Item Insert)
- ORDDSCNT.PC (Order Discount Calculation)
- PRECOSTCALC.PC (Pre-cost Calculation)
- PREPOST.PC (Prepost Functionality for Multi-threadable Programs)
- VENDINVC.PC (Vendor Deal Invoicing)
- VENDINVF.PC (Vendor Deal Invoicing)

## costcalc (Deal Cost Calculation)

### Functional Area

Deals maintenance

### Module Affected

COSTCALC.PC

### Design Overview

This module is responsible for maintaining information on the FUTURE\_COST table based on records on the DEAL\_SKU\_TEMP table.

This batch program calculates the net cost, net net cost, and dead net net cost for all items that are on the DEAL\_SKU\_TEMP table (which contains all items or items in hierarchies on deals that are on the DEAL\_QUEUE table). This contains deals that are about to be approved, unapproved, or closed (any action that would potentially change which deals affect an item). Also items with new item-location relationships, future cost changes or merchandise hierarchy reclassifications or the cancellation of such events appears here. All active deals for each item are used in the calculation along with any future reclassification or cost change information. Once calculated, the costs are inserted into the FUTURE\_COST table.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                 |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 2 (daily)                                                                                                             |
| Scheduling Considerations | Must be run after DITINSRT.PC and PRECOSTCALC.PC.                                                                           |
| Pre-Processing            | N/A                                                                                                                         |
| Post-Processing           | prepost costcalc post                                                                                                       |
| Threading Scheme          | Multiple processes can be run at the same time against different suppliers specified as the input parameter to the program. |

### Restart/Recovery

The logical unit of work for this module is item/supplier/origin country/location/start date on DEAL\_SKU\_TEMP. Since records are deleted from DEAL\_SKU\_TEMP when a commit point is reached, inherent restart/recovery is achieved based on records on the DEAL\_SKU\_TEMP table.

### Locking Strategy

N/A

### Security Considerations

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| PERIOD                 | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS         | Yes    | No     | No     | No     |
| ITEM_LOC               | Yes    | No     | No     | No     |
| DEAL_SKU_TEMP          | Yes    | No     | No     | Yes    |
| SUPS                   | Yes    | No     | No     | No     |
| PACKITEM               | Yes    | No     | No     | No     |
| ITEM_MASTER            | Yes    | No     | No     | No     |
| ITEM_LOC_TEMP          | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY      | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC  | Yes    | No     | No     | No     |
| DEAL_HEAD              | Yes    | No     | No     | No     |
| DEAL_DETAIL            | Yes    | No     | No     | No     |
| DEAL_ITEMLOC           | Yes    | No     | No     | No     |
| DEAL_THRESHOLD         | Yes    | No     | No     | No     |
| DEAL_SUPPLIER_TEMP     | Yes    | Yes    | No     | No     |
| RECLASS_COST_CHG_QUEUE | Yes    | No     | No     | No     |
| FUTURE_COST            | No     | Yes    | No     | Yes    |
| WH                     | Yes    | No     | No     | No     |

**I/O Specification**

N/A

## dealact (Deal Actuals)

### Functional Area

Deals maintenance

### Module Affected

DEALACT.PC

### Design Overview

This program runs on a daily basis and calculates actuals information to update the deal actuals table at the item/location level for bill back non rebate deals, bill back purchase order rebate deals and bill back sales and receipts deals.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                  |
|---------------------------|--------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3, daily                                                                                               |
| Scheduling Considerations | Must be run daily after SALSTAGE.PC; otherwise data is lost and income cannot be calculated retrospectively. |
| Pre-Processing            | prepost dealact_nor pre<br>Prepost_dealact_po_pre<br>Prepost_dealact_sales pre                               |
| Post-Processing           | N/A                                                                                                          |
| Threading Scheme          | Multithreaded on Deal ID                                                                                     |

### Restart/Recovery

The logical unit of work is combination of deal\_id/deal\_detail\_id. The database commit takes place when the number of deal\_id/deal\_detail\_id records processed is equal to commit max counter in the restart control table.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| DEAL_HEAD              | Yes    | No     | No     | No     |
| DEAL_BB_NO_REBATE_TEMP | Yes    | No     | No     | No     |
| DEAL_BB_REBATE_PO_TEMP | Yes    | No     | No     | No     |
| DEAL_TRAN_DATA_TEMP    | Yes    | No     | No     | No     |
| DEAL_ACTUALS_ITEM_LOC  | No     | Yes    | Yes    | No     |

**I/O Specification**

N/A

**dealcls (Deal Close)****Functional Area**

Deals maintenance

**Module Affected**

DEALCLS.PC

**Design Overview**

The purpose of this module is to close any active deals that have reached their close date. It sets the status of “Off Invoice” and “Bill Back” deals to Closed. For “Off Invoice” deals, records are closed on the specified deal close date or when the deal close date has passed, where this module is not run daily.

**Scheduling Constraints**

| Schedule Information      | Description    |
|---------------------------|----------------|
| Processing Cycle          | Phase 3, Daily |
| Scheduling Considerations | N/A            |
| Pre-Processing            | N/A            |
| Post-Processing           | N/A            |
| Threading Scheme          | N/A            |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table      | Select | Insert | Update | Delete |
|------------|--------|--------|--------|--------|
| DEAL_HEAD  | Yes    | No     | Yes    | No     |
| DEAL_QUEUE | Yes    | Yes    | No     | No     |

**I/O Specification**

N/A

**dealday (Deal Tran Data Extract Summed Daily)****Functional Area**

Deals maintenance

**Module Affected**

DEALDAY.PC

**Design Overview**

This batch module creates/updates daily data for deal income sales and purchases. This program extracts data inserted by DEALINC.PC. In order to simplify this program, a dealday pre function (in prepost.pc) sums up the data into a temporary table. A dealday post function (in prepost.pc) copies data to transaction table and then purges temporary tables.

**Scheduling Constraints**

| Schedule Information      | Description                                       |
|---------------------------|---------------------------------------------------|
| Processing Cycle          | Phase 3, Monthly                                  |
| Scheduling Considerations | Should be run after DEALINC.PC and before SALMTH. |
| Pre-Processing            | Prepost .pc with parameters dealday pre           |
| Post-Processing           | Prepost .pc with parameters dealday post          |
| Threading Scheme          | Multithreaded on Location                         |

**Restart/Recovery**

The logical unit of work is a transaction comprising the dept/class/subclass. A commit takes place after the number of dept/class/subclass records processed is greater than or equal to the max counter from the RESTART\_CONTROL table.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| TEMP_TRAN_DATA_SUM | Yes    | No     | No     | No     |
| DAILY_DATA         | Yes    | Yes    | Yes    | No     |
| STORE              | Yes    | No     | No     | No     |
| WH                 | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**dealex (Deal Explode)****Functional Area**

Deals maintenance

**Module Affected**

DEALEX.PC

**Design Overview**

This batch explodes bill-back deals to their item-location components. It runs every night before DEALINC.PC for records related to approved deals that are not at the item location level, and explodes them to the item/location level. In addition this batch also needs to take into account any changes in item hierarchy location relationships since records are written here when a new item is approved, reclassified, or associated with a new item supplier country location.

**Scheduling Constraints**

| Schedule Information      | Description                                                                    |
|---------------------------|--------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3, daily                                                                 |
| Scheduling Considerations | This program must be run before DEALINC and RECLSDLY and after PRECOSTCALC.PC. |
| Pre-Processing            | N/A                                                                            |
| Post-Processing           | N/A                                                                            |
| Threading Scheme          | Multithreaded by Deal ID                                                       |



**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

There is a performance concern regarding the size of the insert/select statement. Due to the structure of the DEAL\_ITEMLOC table the statement needs to take into account all levels of organization hierarchy and merchandise hierarchy, and also deal exclusions. In order to enhance performance inline view DIMIT is used as data lookups for the statements. These tables provide existing item/location and item/merch records as well as new item/location and reclassified items. The complete statement is used by an 'insert' so as to maximize the performance and the dataset is multithreaded by DEAL\_ID.

**Key Tables Affected**

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| DEAL_ITEM_LOC_EXPLODE  | Yes    | Yes    | No     | No     |
| ITEM_LOC               | Yes    | No     | No     | No     |
| GROUPS                 | Yes    | No     | No     | No     |
| DEPS                   | Yes    | No     | No     | No     |
| ITEM_MASTER            | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY      | Yes    | No     | No     | No     |
| WH                     | Yes    | No     | No     | No     |
| STORE_HIERARCHY        | Yes    | No     | No     | No     |
| DEAL_HEAD              | Yes    | No     | No     | No     |
| DEAL_ITEMLOC           | Yes    | No     | No     | No     |
| RECLASS_TRIGGER_TEMP   | Yes    | No     | No     | No     |
| RECLASS_COST_CHG_QUEUE | Yes    | No     | No     | No     |

**I/O Specification**

N/A

## dealfct (Deal Forecast)

### Functional Area

Deals maintenance

### Module Affected

DEALFCT.PC

### Design Overview

This program aggregates income for each item/location and recalculates forecasted values. It maintains forecast periods, deal component totals and deal totals. After determining which active deals need to have forecast periods updated with actuals, the program then sums up all the actuals for the deal reporting period and updates the table with the summed values and change the period from a forecast period to a fixed period. The program also adjusts either the deal component totals or the remaining forecast periods to ensure that the deal totals remain correct. For each deal, the program also maintains values held at header level.

### Scheduling Constraints

| Schedule Information      | Description                                                   |
|---------------------------|---------------------------------------------------------------|
| Processing Cycle          | Phase 3, Ad Hoc                                               |
| Scheduling Considerations | After DEALINC.PC and before SALMONTH.PC.                      |
| Pre-Processing            | prepost dealfct pre – build records in the DEALFCT_TEMP table |
| Post-Processing           | N/A                                                           |
| Threading Scheme          | Threaded by deal ID                                           |

### Restart/Recovery

The logical unit of work is a Deal ID. A commit takes place after the number of deals records processed is equal to the commit max counter from the RESTART\_CONTROL table.

### Locking Strategy

DEAL\_DETAIL table is being locked by the module.

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| DEALFCT_TEMP          | Yes    | No     | No     | No     |
| DEAL_ACTUALS_FORECAST | Yes    | No     | Yes    | No     |
| DEAL_HEAD             | Yes    | No     | Yes    | No     |
| DEAL_DETAIL           | Yes    | No     | Yes    | No     |

**I/O Specification**

N/A

**dealinc (Deal Income Calculation)****Functional Area**

Deals Maintenance

**Module Affected**

DEALINC.PC

**Design Overview**

This program generates incode for each item/location for bill-back deals. It retrieves deal attributes and actuals data from the deals tables for complex deals. It then calculates the income and updates the actuals table with the calculated income value. Additionally the program inserts the income value into the TEMP\_TRAN\_DATA table using the tran types deal sales and deal purchases.

Subsequent programs runs to perform forecast processing for active deals and rolls up TEMP\_TRAN\_DATA rows inserted by the multiple instances of this module and inserts/updates DAILY\_DATA with the summed values and then inserts details from TEMP\_TRAN\_DATA into TRAN\_DATA. Income is calculated by retrieving threshold details for each deal component and determines how to perform the calculation (that is, Linear/Scalar, Actuals Earned/Pro-Rate).

**Scheduling Constraints**

| Schedule Information      | Description                                    |
|---------------------------|------------------------------------------------|
| Processing Cycle          | Phase 3, Monthly                               |
| Scheduling Considerations | Must be run before SALMTH.PC, after DEALACT.PC |
| Pre-Processing            | prepost dealinc pre                            |
| Post-Processing           | N/A                                            |
| Threading Scheme          | Threaded by deal ID                            |

**Restart/Recovery**

The logical unit of work is a DEAL\_ID, DEAL\_DETAIL\_ID combination. A commit takes place after the number of deals records processed is equal to the commit max counter from the RESTART\_CONTROL table.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| DEAL_HEAD             | Yes    | No     | No     | No     |
| DEAL_DETAIL           | Yes    | No     | No     | No     |
| DEAL_ACTUALS_FORECAST | Yes    | No     | No     | No     |
| GTT_DEALINC_DEALS     | Yes    | Yes    | No     | Yes    |
| DEAL_ACTUALS_ITEM_LOC | Yes    | No     | Yes    | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| STORE                 | Yes    | No     | No     | No     |
| WH                    | Yes    | No     | No     | No     |
| TEMP_TRAN_DATA        | No     | Yes    | No     | No     |

**I/O Specification**

N/A

**dealprg (Deals Purge)****Functional Area**

Deals maintenance

**Module Affected**

DEALPRG.PC

**Design Overview**

The purpose of this batch program is to purge deals after they are held in the system for the specified number of history months after they are closed. The batch program also deletes deal performance tables based on the specified number of history months. This program does not cover PO-specific deals, which is purged with the PO.

**Scheduling Constraints**

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad Hoc      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

**Restart/Recovery**

This program has inherent restart/recovery since records that were processed are deleted from the table. As a result, the driving cursor never fetches the same records again.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| DEAL_HEAD              | Yes    | No     | No     | Yes    |
| ORDHEAD_DISCOUNT       | Yes    | No     | No     | No     |
| ORDLOC_DISCOUNT        | Yes    | No     | No     | No     |
| FIXED_DEAL             | Yes    | No     | No     | Yes    |
| DEAL_ACTUALS_ITEM_LOC  | No     | No     | No     | Yes    |
| DEAL_ITEM_LOC_EXPLODE  | No     | No     | No     | Yes    |
| FUTURE_COST            | Yes    | No     | No     | Yes    |
| RECLASS_COST_CHG_QUEUE | No     | No     | No     | Yes    |
| DEAL ACTUALS_FORECAST  | No     | No     | No     | Yes    |
| DEAL_PROM              | No     | No     | No     | Yes    |
| DEAL_THRESHOLD_REV     | No     | No     | No     | Yes    |
| DEAL_QUEUE             | No     | No     | No     | Yes    |
| DEAL_ITEMLOC           | No     | No     | No     | Yes    |
| POP_TERMS_FULFILLMENT  | No     | No     | No     | Yes    |

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| POP_TERMS_DEF        | No     | No     | No     | Yes    |
| DEAL_DETAIL          | No     | No     | No     | Yes    |
| FIXED_DEAL_MERCH_LOC | No     | No     | No     | Yes    |
| FIXED_DEAL_MERCH     | No     | No     | No     | Yes    |
| FIXED_DEAL_DATES     | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**dealupld (Deal Upload)****Functional Area**

Deals maintenance

**Module Affected**

DEALUPLD.PC

**Design Overview**

The DEALUPLD program is used to upload the deals from an input file, which is created by an external system, into RMS. The external program translates the contents of the EDI upload file to RMS values (that is, its location's DUNS number to its RMS number) and also converts the file into RMS format.

**Scheduling Constraints**

| Schedule Information | Description                                                          |
|----------------------|----------------------------------------------------------------------|
| Processing Cycle     | Phase 0 (daily)                                                      |
| Scheduling Diagram   | This program should run as the first batch in the Deals batch cycle. |
| Pre-processing       | N/A                                                                  |
| Post-Processing      | N/A                                                                  |

**Restart/Recovery**

The program uses File based restart recovery process. The logical unit of work is a single deal head detail record and its associated component records in the input file.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| ORDHEAD           | Yes    | No     | No     | No     |
| SUPS              | Yes    | No     | No     | No     |
| UOM_CLASS         | Yes    | No     | No     | No     |
| DEAL_COMP_TYPE    | Yes    | No     | No     | No     |
| DEPS              | Yes    | No     | No     | No     |
| GROUPS            | Yes    | No     | No     | No     |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| ITEM_SUPPLIER     | Yes    | No     | No     | No     |
| STORE             | Yes    | No     | No     | No     |
| DISTRICT          | Yes    | No     | No     | No     |
| REGION            | Yes    | No     | No     | No     |
| AREA              | Yes    | No     | No     | No     |
| CHAIN             | Yes    | No     | No     | No     |
| WH                | Yes    | No     | No     | No     |
| LOC_LIST_HEAD     | Yes    | No     | No     | No     |
| LOC_LIST_DETAIL   | Yes    | No     | No     | No     |
| COUNTRY           | Yes    | No     | No     | No     |
| PACKITEM_BREAKOUT | Yes    | No     | No     | No     |
| PACKITEM          | Yes    | No     | No     | No     |
| CODE_DETAIL       | Yes    | No     | No     | No     |
| DEAL_HEAD         | No     | Yes    | No     | No     |
| DEAL_DETAIL       | No     | Yes    | No     | No     |
| DEAL_ITEM_LOC     | No     | Yes    | No     | No     |
| POP_TERMS_DEF     | No     | Yes    | No     | No     |
| DEAL_THRESHOLD    | No     | Yes    | No     | No     |

## I/O Specification

### Input File Layout

The input file structure should be as below:

FHEAD

```
{
    THEAD      of DHDTL    REQUIRED    for deal head record
        TDETL      REQUIRED    1 deal head record
    TTAIL      REQUIRED    end of deal head record
    THEAD      of DCDTL    REQUIRED    for deal component records
    [
        TDETL      OPTIONAL   for deal component records
    ]
    TTAIL      REQUIRED    end of deal component records
    THEAD      of DIDTL    REQUIRED    for item-loc records
    [
        TDETL      OPTIONAL   for item-loc records
    ]
    TTAIL      REQUIRED    end of item-loc records
    THEAD of PPDTL    REQUIRED    for proof of performance records
    [
        TDETL      OPTIONAL   for proof of performance records
    ]
    TTAIL      REQUIRED    end of proof of performance records
    THEAD      of DTDTL    REQUIRED    for threshold records
    [
        TDETL      OPTIONAL   for threshold records
    ]
    TTAIL      REQUIRED    end of threshold records
}
FTAIL
```



| Record Name | Field Name                     | Field Type     | Default Value                            | Description/Constraints                                                                                                                               |
|-------------|--------------------------------|----------------|------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| FHEAD       | File Type Record Descriptor    | Char(5)        | FHEAD                                    | Identifies file record type (the beginning of the input file).                                                                                        |
|             | File Line Identifier           | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                        |
|             | File Type Definition           | Char(5)        | EDIDU                                    | Identifies file as 'EDI Deals Upload'                                                                                                                 |
|             | File Create Date               | Char(14)       | Create date                              | Current date, formatted to 'YYYYMMDDHH24MISS'.                                                                                                        |
| THEAD       | File Type Record Descriptor    | Char(5)        | THEAD                                    | Identifies file record type to upload a new deal header.                                                                                              |
|             | File Line Identifier           | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                        |
|             | Transaction Detail Record Type | Char(5)        | DHDTL                                    | Identifies file record type Deal Header. This record MUST BE FOLLOWED BY ONE AND ONLY ONE REQUIRED TDETL RECORD that holds the deal head information. |
| TDETL       | File Type Record Descriptor    | Char(5)        | TDETL                                    | Identifies file record type to upload a new deal.                                                                                                     |
|             | File Line Identifier           | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                        |

| Record Name | Field Name   | Field Type  | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-------------|--------------|-------------|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Partner Type | Char(6)     | REQUIRE D                      | Type of the partner the deal applies to. Valid values are 'S' for a supplier, 'S1' for supplier hierarchy level 1 (for example, the manufacturer), 'S2' for supplier hierarchy level 2 (for example, the distributor) and 'S3' for supplier hierarchy level 3 (that is, the wholesaler). Descriptions of these codes are held on the codes table under a code_type of 'SUHL'.<br>Information pertaining to a single deal has to belong to the same supplier, since a deal may have only one supplier hierarchy associated with it. Only items with the same supplier hierarchy can be on the same deal. Supplier hierarchy is stored at an item / supplier / country / location level. |
|             | Partner Id   | Char(10)    | Blank (space character string) | Level of supplier hierarchy (for example, manufacturer, distributor or wholesaler), set up as a partner in the PARTNER table, used for assigning rebates by a level other than supplier. Rebates at this level include all eligible supplier/item/country records assigned to this supplier hierarchy level.<br><br>This field is required if the Partner Type field was set to 'S1', 'S2' or 'S3'. This field must be blank if the Partner Type field was set to 'S'.                                                                                                                                                                                                                 |
|             | Supplier     | Number (10) | Blank (space character string) | Deal supplier's number. This supplier can be at any level of supplier hierarchy.<br><br>This field is required if the Partner Type field was set to 'S'. This field must be blank if the Partner Type field was set to 'S1', 'S2' or 'S3'.                                                                                                                                                                                                                                                                                                                                                                                                                                             |

| Record Name | Field Name                  | Field Type | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                         |
|-------------|-----------------------------|------------|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Type                        | Char(6)    | REQUIRE D                      | Type of the deal. Valid values are A for annual deal, P for promotional deal, O for PO-specific deal or M for vendor-funded markdown. Deal types are held on the codes table under a code type of 'DLHT'.                                                                                                       |
|             | Currency Code               | Char(3)    | Blank (space character string) | Currency code of the deal's currency. All costs on the deal are held in this currency.<br>If Type is 'O', 'P' or 'A', then Currency Code may not be blank. Currency Code has to be blank if Type is 'M'.                                                                                                        |
|             | Active Date                 | Char(14)   | REQUIRE D                      | Date the deal is active. This date determines when deal components begin to be factored into item costs. For a PO-specific deal, the active_date is the order's written date.                                                                                                                                   |
|             | Close Date                  | Char(14)   | Blank (space character string) | Date the deal ends. This date determines when deal components are no longer factored into item costs. It is optional for annual deals, required for promotional deals. It is left NULL for PO-specific deals.<br>Close Date must not be blank if Type is 'P' or 'M'. Close Date has to be blank if Type is 'O'. |
|             | External Reference Number   | Char(30)   | Blank (space character string) | Any given external reference number that is associated with the deal.                                                                                                                                                                                                                                           |
|             | Order Number                | Number (8) | Blank (space character string) | Order the deal applies to, if the deal is PO-specific.                                                                                                                                                                                                                                                          |
|             | Recalculate Approved Orders | Char(1)    | REQUIRE D                      | Indicates if approved orders should be recalculated based on this deal once the deal is approved. Valid values are Y for yes or N for no. Valid values are 'Y' and 'N'.                                                                                                                                         |

| Record Name | Field Name              | Field Type  | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                                                      |
|-------------|-------------------------|-------------|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Comments                | Char (2000) | Blank (space character string) | Free-form comments entered with the deal.                                                                                                                                                                                                                                                                                                                    |
|             | Billing Type            | Char(6)     | REQUIRE D                      | Billing type of the deal component. Valid values are 'OI' for off-invoice, 'BB' for bill-back, 'BBVFP' for vendor funded promotion and 'BBVFM' for vendor funded markdown. Billing types are held on the codes table under a code type of 'DLBT'.                                                                                                            |
|             | Bill Back Period        | Char(6)     | Blank (space character string) | Code that identifies the bill-back period for the deal component. This field only populates for billing types of 'BB' or 'BBVFP' or 'BBVFM'. Valid bill back period codes are 'W', 'M', 'Q', 'H', 'A'.<br><br>If Billing Type is 'BB' then Bill Back Period must not be blank; if Billing Type is 'OI' (off invoice), then Bill back Period has to be blank. |
|             | Deal Application Timing | Char(6)     | Blank (space character string) | Indicates when the deal component should be applied - at PO approval or time of receiving. Valid values are 'O' for PO approval, 'R' for receiving. These values are held on the codes tables under a code type of 'AALC'. It must be NULL for an M-type deal (vendor funded markdown).                                                                      |

| Record Name | Field Name                      | Field Type | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------|---------------------------------|------------|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Threshold Limit Type            | Char(6)    | Blank (space character string) | Identifies whether thresholds are set up as qty values, currency amount values or percentages (growth rebates only). Valid values are 'Q' for qty, 'A' for currency amount. Threshold limit types are held on the codes table under a code type of 'DLLT'. It must be NULL for an M-type deal (vendor funded markdown) or if the threshold value type is 'Q' (buy/get deals).<br>If Growth Rebate Indicator is 'Y', then the Threshold Limit Type has to be 'Q', 'A' or NULL.                                            |
|             | Threshold Limit Unit of Measure | Char(4)    | Blank (space character string) | Unit of measure of the threshold limits, if the limit type is quantity. Only Unit of Measures with a UOM class of 'VOL' (volume), 'MASS' or 'QTY' (quantity) can be used in this field. Valid Unit of Measures can be found on the UOM_CLASS table.<br>If the Threshold Limit Type is 'A', then Threshold Limit Unit of Measure has to be blank. If the Threshold Limit Type is 'Q', Threshold Limit Unit of Measure must not be blank. If Threshold Limit Type is blank, Threshold Limit Unit of Measure must be blank. |
|             | Rebate Indicator                | Char(1)    | REQUIRE D                      | Indicates if the deal component is a rebate. Deal components can only be rebates for bill-back billing types. Valid values are 'Y' for yes or 'N' for no.<br>If Billing Type is 'OI', then Rebate Indicator must be 'N'.                                                                                                                                                                                                                                                                                                 |

| Record Name | Field Name                       | Field Type | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                           |
|-------------|----------------------------------|------------|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Rebate Calculation Type          | Char(6)    | Blank (space character string) | Indicates if the rebate should be calculated using linear or scalar calculation methods. Valid values are 'L' for linear or 'S' for scalar. This field is required if the rebate indicator is 'Y'. Rebate calculation types are held on the codes table under a code type of 'DLCT'.<br>If Rebate Indicator is 'Y', then Rebate Calculation Type must not be blank. Otherwise it has to be blank. |
|             | Growth Rebate Indicator          | Char(1)    | REQUIRE D                      | Indicates if the rebate is a growth rebate, meaning it is calculated and applied based on an increase in purchases or sales over a specified period of time. Valid values are 'Y' for yes or 'N' for no.<br>If Rebate Indicator is 'N', then Growth Rebate Indicator must be 'N'.                                                                                                                 |
|             | Historical Comparison Start Date | Char(14)   | Blank (space character string) | The first date of the historical period against which growth is measured in this growth rebate. Note performance and the rebate amount are not calculated - this field is for informational/reporting purposes only.<br>If Growth Rebate Indicator is 'Y', then Historical Comparison Start Date must not be blank. Otherwise it must be blank.                                                   |
|             | Historical Comparison End Date   | Char(14)   | Blank (space character string) | The last date of the historical period against which growth is measured in this growth rebate. Note performance and the rebate amount are not calculated - this field is for informational/reporting purposes only.<br>If Growth Rebate Indicator is 'Y', then Historical Comparison End Date must not be blank. Otherwise it must be blank.                                                      |

| Record Name | Field Name                                      | Field Type     | Default Value                            | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------|-------------------------------------------------|----------------|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Rebate Purchases or Sales Application Indicator | Char(6)        | Blank (space character string)           | Indicates if the rebate should be applied to purchases or sales. Valid values are 'P' for purchases or 'S' for sales. It is required if the rebate indicator is 'Y'. Rebate purchase/sales indicators are held on the codes table under a code type of 'DLRP'.<br><br>If the Rebate Indicator is 'Y', then the Rebate Purchases or Sales Application Indicator must not be blank. Otherwise it has to be blank. |
|             | Security Indicator                              | Char           | Y                                        | Security Indicator                                                                                                                                                                                                                                                                                                                                                                                              |
| TTAIL       | File Line Identifier                            | Char(5)        | TTAIL                                    | Identifies file record type (the end of the transaction detail).                                                                                                                                                                                                                                                                                                                                                |
|             | File Line Identifier                            | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                                                                                                                                                  |
|             | Transaction Record Counter                      | Numeric ID(6)  | Sequential number<br>Created by program. | Number of records/transactions in current transaction set (only records between thead and ttail). For DHDTL TDETL records this is always 1!                                                                                                                                                                                                                                                                     |
| THEAD       | File Type Record Descriptor                     | Char(5)        | THEAD                                    | Identifies file record type to upload a new deal sub loop.                                                                                                                                                                                                                                                                                                                                                      |
|             | File Line Identifier                            | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                                                                                                                                                  |
|             | Transaction Detail Record Type                  | Char(5)        | DCDTL                                    | Identifies file record type of sub loop as Deal Component Detail.                                                                                                                                                                                                                                                                                                                                               |
| TDETL       | File Type Record Descriptor                     | Char(5)        | TDETL                                    | Identifies file record type to upload deal components.                                                                                                                                                                                                                                                                                                                                                          |

| Record Name | Field Name                       | Field Type     | Default Value                            | Description/Constraints                                                                                                                                                                                                                                                                    |
|-------------|----------------------------------|----------------|------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | File Line Identifier             | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                             |
|             | Deal Component Type              | Char(6)        | REQUIRE D                                | Type of the deal component, user-defined and stored on the DEAL_COMP_TYPE table.                                                                                                                                                                                                           |
|             | Application Order                | Number (10)    | Blank (space character string)           | Number indicating the order in which the deal component should be applied with respect to any other deal components applicable to the item within the deal. This number is unique across all deal components within the deal. It must be NULL for an M-type deal (vendor funded markdown). |
|             | Collect Start Date               | Char(14)       | Blank (space character string)           | Date that collection of the bill-back should begin.<br>If Billing Type is 'BB' then Collect Start Date must not be blank, otherwise it has to be blank.                                                                                                                                    |
|             | Collect End Date                 | Char(14)       | Blank (space character string)           | Date that collection of the bill-back should end.<br>If Billing Type is 'BB' then Collect End Date must not be blank, otherwise it has to be blank.                                                                                                                                        |
|             | Cost Application Level Indicator | Char(6)        | Blank (space character string)           | Indicates what cost bucket the deal component should affect. Valid values are 'N' for net cost, 'NN' for net cost and 'DNN' for dead net cost. These values are held on the codes tables under a code type of 'DLCA'. It must be NULL for an M-type deal (vendor funded markdown).         |
|             | Pricing Cost Indicator           | Char(1)        | REQUIRE D                                | Identifies deal components that should be included when calculating a pricing cost.<br>Valid values are 'Y'es and 'N'o.                                                                                                                                                                    |



| Record Name | Field Name           | Field Type | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-------------|----------------------|------------|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Deal Class           | Char(6)    | Blank (space character string) | Identifies the calculation class of the deal component. Valid values are 'CU' for cumulative (discounts are added together and taken off as one lump sum), 'CS' for cascade (discounts are taken one at a time with subsequent discounts taken off the result of the previous discount) and 'EX' for exclusive (overrides all other discounts). 'EX' type deal components are only valid for promotional deals. Deal classes are held on the codes table under a code type of 'DLCL'. It must be NULL for an M-type deal (vendor funded markdown).                     |
|             | Threshold Value Type | Char(6)    | Blank (space character string) | Identifies whether the discount values associated with the thresholds are set up as qty values, currency amount values, percentages or fixed amounts. Valid values are 'Q' for qty, 'A' for currency amount, 'P' for percentage or 'F' for fixed amount. Qty threshold value (buy/get) deals are only allowed on off-invoice discounts. Deal threshold value types are held on the codes table under a code type of 'DLL2'. It must be NULL for an M-type deal (vendor funded markdown).<br>If Billing Type is 'BB', then the Threshold Value Type must be 'A' or 'P'. |
|             | Buy Item             | Char(25)   | Blank (space character string) | Identifies the item that must be purchased for a quantity threshold-type discount. This value is required for quantity threshold value type discounts. Otherwise it has to be blank.                                                                                                                                                                                                                                                                                                                                                                                   |

| Record Name | Field Name          | Field Type   | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-------------|---------------------|--------------|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Get Type            | Char(6)      | Blank (space character string) | Identifies the type of the 'get' discount for a quantity threshold-type (buy/get) discount. Valid values include 'X' (free), 'P' (percent), 'A' (amount) and 'F' (fixed amount). They are held on the codes table under a code type of 'DQGT'. This value is required for quantity threshold value deals. Otherwise it has to be blank.                                                                                                                |
|             | Get Value           | Number(20,4) | All 0s.                        | Identifies the value of the 'get' discount for a quantity threshold-type (buy/get) discount that is not a 'free goods' deal. The Get Type above identifies the type of this value. This value is required for quantity threshold value type deals that are not a Get Type of free. Otherwise it has to be 0.<br><br>If Get Type is 'P', 'A' or 'F', then Get Value must not be blank. If the Get Type is 'X' or blank, then Get Value has to be blank. |
|             | Buy Item Quantity   | Number(12,4) | All 0s.                        | Identifies the quantity of the threshold 'buy' item that must be ordered to qualify for the 'free' item. This value is required for quantity threshold value type discounts. Otherwise it has to be 0.                                                                                                                                                                                                                                                 |
|             | Recursive Indicator | Char(1)      | REQUIRE D                      | For 'buy/get free' discounts, indicates if the quantity threshold discount is only for the first 'buy amt.' purchased (that is, for the first 10 purchased, get 1 free), or if a free item is given for every multiple of the 'buy amt' purchased on the order (that is, for each 10 purchased, get 1 free). Valid values are 'Y' for yes or 'N' for no.<br><br>If the Get Type is blank, then Recursive Indicator has to be 'N'.                      |

| Record Name | Field Name                                          | Field Type   | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                                                                               |
|-------------|-----------------------------------------------------|--------------|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Buy Item Order Target Quantity                      | Number(12,4) | All 0s.                        | Indicates the targeted purchase level for all locations on a purchase order. This is the target level that is used for future calculation of net cost. This value is required for quantity threshold value type deals. Otherwise it has to be 0.                                                                                                                                      |
|             | Average Buy Item Order Target Quantity Per Location | Number(12,4) | All 0s.                        | Indicates the average targeted purchase level per location on the deal. This value is used in future cost calculations. This value is required for quantity threshold value type deals. Otherwise it has to be 0.                                                                                                                                                                     |
|             | Get Item                                            | Char(25)     | Blank (space character string) | Identifies the 'get' item for a quantity threshold-type (buy/get) discount. This value is required for quantity threshold value deals. Otherwise it has to be blank.<br><br>If Get Type is 'P', 'A', 'F' or 'X', then Get Item must not be blank. If the Get Type is blank, then Get Item has to be blank.                                                                            |
|             | Get Quantity                                        | Number(12,4) | All 0s.                        | Identifies the quantity of the identified 'get' item that is given at the specified 'get' discount if the 'buy amt' of the buy item is purchased. This value is required for quantity threshold value type discounts. Otherwise it has to be 0.<br><br>If Get Type is 'P', 'A', 'F' or 'X', then Get Quantity must not be 0. If the Get Type is blank, then Get Quantity has to be 0. |

| Record Name | Field Name                           | Field Type     | Default Value                            | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-------------|--------------------------------------|----------------|------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Free Item Unit Cost                  | Number(20,4)   | All 0s.                                  | For 'buy/get free' discounts, identifies the unit cost of the threshold 'free' item that is used in calculating the prorated qty. discount. It defaults to the item/supplier cost, but can be modified based on the agreement with the supplier. It must be greater than zero as this is the cost that would normally be charged for the goods if no deal applied.<br>If Get Type is 'P', 'A', 'F' or blank, then Free Item Unit Cost must be 0. If the Get Type is 'X', then Free Item Unit Cost must not be 0. |
|             | Transaction Level Discount Indicator | Char(1)        | REQUIRE D                                | Indicates if the discount is a transaction-level discount (that is, 10% across an entire PO).<br>Valid Values are 'Y' or 'N'. If set to 'Y', Deal Class has to be 'CU' and Billing Type has to be 'OI'. No DIDTL or PPDTL records may be present for a Transaction Level Discount DCDTL record.                                                                                                                                                                                                                  |
|             | Comments                             | Char(2000)     | Blank (space character string)           | Free-form comments entered with the deal component.                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| TTAIL       | File Line Identifier                 | Char(5)        | TTAIL                                    | Identifies file record type (the end of the transaction detail).                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|             | File Line Identifier                 | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|             | Transaction Record Counter           | Numeric ID(6)  | Sequential number<br>Created by program. | Number of records/transactions in current transaction set (only records between thead and ttail)                                                                                                                                                                                                                                                                                                                                                                                                                 |
| THEAD       | File Type Record Descriptor          | Char(5)        | THEAD                                    | Identifies file record type to upload a new deal sub loop.                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

| Record Name | Field Name                     | Field Type     | Default Value                            | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------|--------------------------------|----------------|------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | File Line Identifier           | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                                                                                                                                                                                  |
|             | Transaction Detail Record Type | Char(5)        | DIDTL                                    | Identifies file record type of sub loop as Deal Component Item-location Detail.                                                                                                                                                                                                                                                                                                                                                                 |
| TDETL       | File Type Record Descriptor    | Char(5)        | TDETL                                    | Identifies file record type to upload deal item-location details.                                                                                                                                                                                                                                                                                                                                                                               |
|             | File Line Identifier           | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                                                                                                                                                                                  |
|             | Merchandise Level              | Char(6)        | REQUIRE D                                | Indicates what level of the merchandise hierarchy the record is at. Valid values include '1' for company-wide (all items), '2' for division, '3' for group, '4' for dept, '5' for class, '6' for subclass, '7' for line, '8' for line/differentiator 1, '9' for line/differentiator 2, '10' for line/differentiator 3, '11' for line/differentiator 4 and '12' for . These level types are held on the codes table under a code type of 'DIML'. |
|             | Company Indicator              | Char(1)        | REQUIRE D                                | Indicates if the deal component is applied company-wide (that is, whether all items in the system are included in the discount or rebate). Valid values are 'Y' for yes and 'N' for no.                                                                                                                                                                                                                                                         |
|             | Division                       | Number (4)     | Blank (space character string).          | ID of the division included in or excluded from the deal component. Valid values are on the DIVISION table.<br><br>If Group is not blank, then Division must not be blank. If Merchandise Level is 2, then Division must not be blank and Group, Department, Class and Subclass must be blank.                                                                                                                                                  |

| Record Name | Field Name  | Field Type | Default Value                   | Description/Constraints                                                                                                                                                                                                                                                                            |
|-------------|-------------|------------|---------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Group       | Number (4) | Blank (space character string). | ID of the group included in or excluded from the deal component. Valid values are on the GROUPS table.<br><br>If Department is not blank, then Group must not be blank. If Merchandise Level is 3, then Group must not be blank and Department, Class and Subclass must be blank.                  |
|             | Department  | Number (4) | Blank (space character string). | ID of the department included in or excluded from the deal component. Valid values are on the DEPS table.<br><br>If Class is not blank, then Department must not be blank. If Merchandise Level is 4, then Department must not be blank and Class and Subclass must be blank.                      |
|             | Class       | Number (4) | Blank (space character string). | ID of the class included in or excluded from the deal component. Valid values are on the CLASS table.<br><br>If Subclass is not blank, then Class must not be blank. If Merchandise Level is 5, then Class must not be blank and Subclass must be blank.                                           |
|             | Subclass    | Number (4) | Blank (space character string). | ID of the subclass included in or excluded from the deal component. Valid values are on the SUBCLASS table.<br><br>If Merchandise Level is 6 or more than 6, then Subclass must not be blank.                                                                                                      |
|             | Item Parent | Char(25)   | Blank (space character string)  | Alphanumeric value that uniquely identifies the item/group at the level above the item. This value must exist as an item in another row on the ITEM_MASTER table.<br><br>If Merchandise Level is 7, then Item Parent or Item Grandparent must not be blank (at least one of them has to be given). |

| Record Name | Field Name        | Field Type | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                  |
|-------------|-------------------|------------|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Item Grandparent  | Char(25)   | Blank (space character string) | Alphanumeric value that uniquely identifies the item/group two levels above the item. This value must exist as both an item and an item parent in another row on the ITEM_MASTER table.<br><br>If Merchandise Level is 7, then Item Parent or Item Grandparent must not be blank (at least one of them has to be given). |
|             | Differentiat or 1 | Char(10)   | Blank (space character string) | Diff_group or diff_id that differentiates the current item from its item_parent.<br><br>If Item Grandparent, Item Parent and Differentiator 2 are blank, then Differentiator 1 must be blank. If Merchandise Level is 8, then Differentiator 1 must not be blank.                                                        |
|             | Differentiat or 2 | Char(10)   | Blank (space character string) | Diff_group or diff_id that differentiates the current item from its item_parent.<br><br>If Item Grandparent, Item Parent and Differentiator 1 are blank, then Differentiator 2 must be blank. If Merchandise Level is 9, then Differentiator 2 must not be blank.                                                        |
|             | Differentiat or 3 | Char(10)   | Blank (space character string) | Diff_group or diff_id that differentiates the current item from its item_parent.<br><br>If Item Grandparent, Item Parent and Differentiator 1 and 2 are blank, then Differentiator 3 must be blank. If Merchandise Level is 10, then Differentiator 3 must not be blank.                                                 |
|             | Differentiat or 4 | Char(10)   | Blank (space character string) | Diff_group or diff_id that differentiates the current item from its item_parent.<br><br>If Item Grandparent, Item Parent and Differentiator 1, 2 and 3 are blank, then Differentiator 4 must be blank. If Merchandise Level is 10, then Differentiator 4 must not be blank.                                              |

| Record Name | Field Name           | Field Type  | Default Value                   | Description/Constraints                                                                                                                                                                                                                                                                                                                                                          |
|-------------|----------------------|-------------|---------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Organizational Level | Char(6)     | Blank (space character string)  | Indicates what level of the organizational hierarchy the record is at. Valid values include '1' for chain, '2' for area, '3' for region, '4' for district and '5' for location. These level types are held on the codes table under a code type of 'DIOL'.<br>If company indicator is N, this must not be blank. If location type is warehouse or location list, this must be 5. |
|             | Chain                | Number (10) | Blank (space character string). | ID of the chain included in or excluded from the deal component. Valid values are on the CHAIN table.<br>If org. level is 1, this field must not be blank.                                                                                                                                                                                                                       |
|             | Area                 | Number (10) | Blank (space character string). | ID of the area included in or excluded from the deal component. Valid values are on the AREA table.<br>If org. level is 2, this field and chain must not be blank.                                                                                                                                                                                                               |
|             | Region               | Number (10) | Blank (space character string). | ID of the region included in or excluded from the deal component. Valid values are on the REGION table.<br>If org. level is 3, this field, area, and chain must not be blank.                                                                                                                                                                                                    |
|             | District             | Number (10) | Blank (space character string). | ID of the district included in or excluded from the deal component. Valid values are on the DISTRICT table.<br>If org. level is 4, then this field, region, area, and chain must not be blank.                                                                                                                                                                                   |



| Record Name | Field Name                | Field Type     | Default Value                            | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------|---------------------------|----------------|------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Location                  | Number (10)    | Blank (space character string).          | ID of the location included in or excluded from the deal component. Valid values are on the STORE, WH, or LOC_LIST_HEAD table.<br>If org. level is 5, this field must not be blank. Chain, area, region, and district should be blank if the loc_type is L or W. If the loc_type is S, then they all must not be blank.<br>If Location Type is not blank, then Location must not be blank. Otherwise it has to be blank. |
|             | Origin Country Identifier | Char(3)        | Blank (space character string)           | Origin country of the item that the deal component should apply to.                                                                                                                                                                                                                                                                                                                                                      |
|             | Location Type             | Char(1)        | Blank (space character string)           | Type of the location referenced in the location field. Valid values are 'S' and 'W'. Location types are held on the codes table under the code type 'LOC3'.<br>If location is blank then this field has to be blank also.                                                                                                                                                                                                |
|             | Item                      | Char(25)       | Blank (space character string)           | Unique alphanumeric value that identifies the item.<br>If Merchandise Level is 10, then Item must not be blank.                                                                                                                                                                                                                                                                                                          |
|             | Exclusion Indicator       | Char(1)        | REQUIRE D                                | Indicates if the deal component item/location line is included in the deal component or excluded from it. Valid values are 'Y' for yes or 'N' for no.                                                                                                                                                                                                                                                                    |
|             | Reference Line            | Number (10)    | REQUIRE D                                | This value determines which line in the input file this item-loc record belongs to.                                                                                                                                                                                                                                                                                                                                      |
| TTAIL       | File Line Identifier      | Char(5)        | TTAIL                                    | Identifies file record type (the end of the transaction detail).                                                                                                                                                                                                                                                                                                                                                         |
|             | File Line Identifier      | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                                                                                                                                                           |

| Record Name | Field Name                     | Field Type     | Default Value                         | Description/Constraints                                                                                                                                                                                                                                                                        |
|-------------|--------------------------------|----------------|---------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Transaction Record Counter     | Numeric ID(6)  | Sequential number Created by program. | Number of records/transactions in current transaction set (only records between thead and ttail)                                                                                                                                                                                               |
| THEAD       | File Type Record Descriptor    | Char(5)        | THEAD                                 | Identifies file record type to upload a new deal sub loop.                                                                                                                                                                                                                                     |
|             | File Line Identifier           | Numeric ID(10) | Sequential number Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                                 |
|             | Transaction Detail Record Type | Char(5)        | PPDTL                                 | Identifies file record type of sub loop as Proof of Performance Detail.                                                                                                                                                                                                                        |
| TDETL       | File Type Record Descriptor    | Char(5)        | TDETL                                 | Identifies file record type to upload deal proof of performance details.                                                                                                                                                                                                                       |
|             | File Line Identifier           | Numeric ID(10) | Sequential number Created by program. | ID of current line being read from input file.                                                                                                                                                                                                                                                 |
|             | Deal Sub Item                  | Char(25)       |                                       | Specific transaction level (or below) item that's proof of performance is being measured. This can be populated when the deal itself is on a case UPC but the proof of performance is on an individual selling unit.                                                                           |
|             | Proof of Performance Type      | Char(6)        | REQUIRE D                             | Code that identifies the proof of performance type (that is, the term is that the item must be displayed on an end cap for 28 days - the pop_type is code 'ECD' for end cap display). Valid values for this field are stored in the code_type = 'PPT'. This field is required by the database. |

| Record Name | Field Name                      | Field Type    | Default Value                  | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------|---------------------------------|---------------|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Proof of Performance Value      | Number (20,4) | All 0s.                        | Value that describes the term of the proof of performance type (that is, the term is that the item must be displayed on an end cap for 28 days - the pop_value is 28). This field is required by the database if the record has a pop_value_type.<br>If Proof of Performance Value is not blank, then Proof of Performance Value Type must not be blank. If Proof of Performance Value is blank, then Proof of Performance Value Type must be blank.                                                                         |
|             | Proof of Performance Value Type | Char(6)       | Blank (space character string) | Value that describes the type of the pop_value (that is, the term is that the item must be displayed on an end cap for 28 days - the pop_value_type is the code 'DAYS' for days). Valid values for this field are stored in the code_type = 'PPVT'. This field is required by the database if the record has a pop_value.<br>If Proof of Performance Value is not blank, then Proof of Performance Value Type must not be blank. If Proof of Performance Value is blank, then Proof of Performance Value Type must be blank. |
|             | Vendor Recommended Start Date   | Char(14)      | Blank (space character string) | This column holds the date that the vendor recommends that the POP begin.                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|             | Vendor Recommended End Date     | Char(14)      | Blank (space character string) | This column holds the date that the vendor recommends that the POP end.                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|             | Planned Start Date              | Char(14)      | Blank (space character string) | This column holds the date that the merchandiser/category manager plans to begin the POP.                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|             | Planned End Date                | Char(14)      | Blank (space character string) | This column holds the date that the merchandiser/category manager plans to end the POP.                                                                                                                                                                                                                                                                                                                                                                                                                                      |

| Record Name | Field Name                     | Field Type     | Default Value                            | Description/Constraints                                                                          |
|-------------|--------------------------------|----------------|------------------------------------------|--------------------------------------------------------------------------------------------------|
|             | Comment                        | Char(255)      | Blank (space character string)           | Free-form comments.                                                                              |
|             | Reference Line                 | Number (10)    | REQUIRE D                                | This value determines which line in the input file this Proof of Performance record belongs to.  |
| TTAIL       | File Line Identifier           | Char(5)        | TTAIL                                    | Identifies file record type (the end of the transaction detail).                                 |
|             | File Line Identifier           | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                   |
|             | Transaction Record Counter     | Numeric ID(6)  | Sequential number<br>Created by program. | Number of records/transactions in current transaction set (only records between thead and ttail) |
| THEAD       | File Type Record Descriptor    | Char(5)        | THEAD                                    | Identifies file record type to upload a new deal sub loop.                                       |
|             | File Line Identifier           | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                   |
|             | Transaction Detail Record Type | Char(5)        | DTDTL                                    | Identifies file record type of sub loop as Deal Component Threshold Detail.                      |
| TDETL       | File Type Record Descriptor    | Char(5)        | TDETL                                    | Identifies file record type to upload deal threshold details.                                    |
|             | File Line Identifier           | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                   |

| Record Name | Field Name             | Field Type    | Default Value | Description/Constraints                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------|------------------------|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Lower Limit            | Number (20,4) | REQUIRE D     | Lower limit of the deal component. This is the minimum value that must be met in order to get the specified discount. This value is either a currency amount or quantity value, depending on the value in the deal_detail.threshold_limit_type field of this deal component (Threshold Value Type field of the DCDTL record that this DTDTL record belongs to as specified in the reference line field). |
|             | Upper Limit            | Number (20,4) | REQUIRE D     | Upper limit of the deal component. This is the maximum value for which the specified discount applies. This value is either a currency amount or quantity value, depending on the value in the deal_detail.threshold_limit_type field of this deal component (Threshold Value Type field of the DCDTL record that this DTDTL record belongs to as specified in the reference line field).                |
|             | Value                  | Number (20,4) | REQUIRE D     | Value of the discount that is given for meeting the specified thresholds for this deal component. This value is either a currency amount or quantity value, depending on the value in the deal_detail.threshold_value_type field of this deal component (Threshold Value Type field of the DCDTL record that this DTDTL record belongs to as specified in the reference line field).                     |
|             | Target Level Indicator | Char(1)       | REQUIRE D     | Indicates if a threshold level is the targeted purchase or sales level for a deal component. This indicator is used for cost calculations. Valid values are 'Y' for yes and 'N' for no.                                                                                                                                                                                                                  |
|             | Reference Line         | Number (10)   | REQUIRE D     | This value determines which line in the input file this Threshold record belongs to.                                                                                                                                                                                                                                                                                                                     |

| Record Name | Field Name                 | Field Type     | Default Value                            | Description/Constraints                                                                          |
|-------------|----------------------------|----------------|------------------------------------------|--------------------------------------------------------------------------------------------------|
| TTAIL       | File Line Identifier       | Char(5)        | TTAIL                                    | Identifies file record type (the end of the transaction detail).                                 |
|             | File Line Identifier       | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                   |
|             | Transaction Record Counter | Numeric ID(6)  | Sequential number<br>Created by program. | Number of records/transactions in current transaction set (only records between thead and ttail) |
| FTAIL       | File Line Identifier       | Char(5)        | FTAIL                                    | Identifies file record type (the end of the input file).                                         |
|             | File Line Identifier       | Numeric ID(10) | Sequential number<br>Created by program. | ID of current line being read from input file.                                                   |
|             | File Record Counter        | Numeric ID(10) | Sequential number<br>Created by program. | Number of records/transactions in current file (only records between head and tail)              |

## discotbapply (Discount OTB Apply)

### Functional Area

OTB

### Module Affected

DISCOTBAPPLY.PC

### Design Overview

This batch program is to be run after orddscnt to communicate the change in costs to open to buy (OTB). The order discount module (ORDDSCNT) inserts into DISC\_OTB\_APPLY table each approved order with items that are on deals. For each dept, class, subclass combination on this table, DISCOTBAPPLY calls build order receive function to update OTB, passing in the difference between the original cost and the final cost. If elc\_ind on system\_options is on and the associated OTB calculation type is 'C' (cost), the difference in cost is updated in order currency; otherwise, the cost difference is converted to the primary currency before updating OTB.

## Scheduling Constraints

| Schedule Information      | Description                                  |
|---------------------------|----------------------------------------------|
| Processing Cycle          | Phase 4 (Daily)                              |
| Scheduling Considerations | This module should be run after ORDDSCNT.PC. |
| Pre-Processing            | N/A                                          |
| Post-Processing           | N/A                                          |
| Threading Scheme          | Multithreaded on department                  |

## Restart/Recovery

This program has inherent restart ability, because records are deleted from DISC\_OTB\_APPLY as they are processed. Array processing is used. Based on the commit\_max\_ctr set on RESTART\_CONTROL table for this batch program, records are array fetched from DISC\_OTB\_APPLY table, processed and committed to the database.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| DISC_OTB_APPLY | Yes    | No     | No     | Yes    |
| ORDHEAD        | Yes    | No     | No     | No     |
| OTB            | No     | No     | Yes    | No     |

## I/O Specification

N/A

## ditinsrt (Deal Item Insert)

### Functional Area

Deals maintenance

### Module Affected

DITINSRT.PC

### Design Overview

This batch program populates the DEAL\_SKU\_TEMP table with all items that are on non vendor-funded, non PO-specific deals listed on the DEAL\_QUEUE table, and all items that fall within a hierarchy from these deals. It gets values for the entire merchandise and organizational hierarchies to populate DEAL\_SKU\_TEMP. The DEAL\_SKU\_TEMP table then uses PRECOSTCALC.PC and COSTCALC.PC to calculate, or recalculate, future costs for all listed items.

In addition, this program populates the deal calculations table (DEAL\_CALC\_QUEUE) with orders that may be affected by non vendor-funded, non PO-specific deals that are on the DEAL\_QUEUE table for future processing by ORDDSCNT.PC. Orders with attached deals that no longer apply are inserted into the DEAL\_CALC\_QUEUE table.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                             |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 1, Daily                                                                                                                          |
| Scheduling Considerations | COSTCALC.PC and ORDDSCNT.PC should run after this module                                                                                |
| Pre-Processing            | N/A                                                                                                                                     |
| Post-Processing           | N/A                                                                                                                                     |
| Threading Scheme          | Multiple processes can be run at the same time against different suppliers or partners specified as the input parameter to the program. |

### Restart/Recovery

The logical unit of work of this program is at the deal header level. A commit occurs when all details of a deal are processed. Inherent restart/recovery is achieved through deleting deals from the DEAL\_QUEUE table when they are processed. Because DEAL\_QUEUE is part of the driving cursor, processed deals are not fetched again when the program restarts.

### Locking Strategy

N/A

### Security Considerations

N/A



**Performance Considerations**

N/A

**Key Tables Affected**

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| DEAL_QUEUE        | Yes    | No     | No     | No     |
| DEAL_ITEMLOC      | Yes    | No     | No     | No     |
| DEAL_HEAD         | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY | Yes    | No     | No     | No     |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| DEPS              | Yes    | No     | No     | No     |
| GROUPS            | Yes    | No     | No     | No     |
| DIVISION          | Yes    | No     | No     | No     |
| CHAIN             | Yes    | No     | No     | No     |
| AREA              | Yes    | No     | No     | No     |
| REGION            | Yes    | No     | No     | No     |
| DISTRICT          | Yes    | No     | No     | No     |
| WH                | Yes    | No     | No     | No     |
| STORE             | Yes    | No     | No     | No     |
| DEAL_SKU_TEMP     | Yes    | Yes    | No     | No     |
| ITEM_LOC          | Yes    | No     | No     | No     |
| ORDHEAD           | Yes    | No     | No     | No     |
| ORDLOC_DISCOUNT   | Yes    | No     | No     | No     |
| DEAL_CALC_QUEUE   | Yes    | Yes    | No     | No     |
| DEAL_QUEUE        | No     | No     | No     | Yes    |

**I/O Specification**

N/A

## orddscent (Order Deal Discount)

### Functional Area

Deals maintenance

### Module Affected

ORDDSCNT.PC

### Design Overview

This module applies deals to a purchase order by calculating the discounts and rebates that are applicable to a purchase order. It fetches orders that need to be recalculated for cost from DEAL\_CALC\_QUEUE. Using the dealordlib shared library, it updates the unit cost and populates ORDLOC\_DISCOUNT and ORDHEAD\_DISCOUNT tables.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                   |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 4, Daily                                                                                                                                |
| Scheduling Considerations | This program should run after DITINSRT. It should run before DISCOTBAPPLY in Phase 4, and before DEALCLS or DEALPRG in the deals batch cycle. |
| Pre-Processing            | N/A                                                                                                                                           |
| Post-Processing           | N/A                                                                                                                                           |
| Threading Scheme          | Multithreaded by supplier                                                                                                                     |

### Restart/Recovery

This program has inherent restart ability, since records are deleted from deal\_calc\_queue as they are processed. Recommended maximum commit counter is low.

### Locking Strategy

This program is attempted to obtain a read lock with a call to check\_lock\_and\_validate function.

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| DISC_OTB_APPLY        | No     | Yes    | No     | No     |
| REV_ORDERS            | No     | Yes    | No     | No     |
| ORD_LC_AMENDMENTS     | No     | Yes    | Yes    | Yes    |
| DEAL_CALC_QUEUE       | Yes    | No     | No     | Yes    |
| ORDHEAD               | Yes    | No     | No     | No     |
| SUPS                  | Yes    | No     | No     | No     |
| CURRENCIES            | Yes    | No     | No     | No     |
| ORDLOC_INVC_COST      | No     | Yes    | Yes    | Yes    |
| ORDLOC                | No     | No     | Yes    | No     |
| ORDLOC_DISCOUNT       | No     | Yes    | Yes    | Yes    |
| ORDHEAD_DISCOUNT      | No     | Yes    | No     | Yes    |
| ORDLOC_DISCOUNT_BUILD | No     | Yes    | No     | Yes    |
| ORD_LC_AMENDMENTS     | No     | Yes    | Yes    | Yes    |

**I/O Specification**

N/A

**precostcalc (Pre-cost Calculation)****Functional Area**

Deal maintenance

**Module Affected**

PRECOSTCALC.PC

**Design Overview**

This batch module is responsible for data maintenance tasks that are necessary before running COSTCALC.

Unprocessed records in RECLASS\_COST\_CHG\_QUEUE drive this program. The driving cursor is a set of seven cursors whose goal is to make sure the DEAL\_SKU\_TEMP table is inserted for three scenarios.

1. If an unprocessed record exists on RECLASS\_COST\_CHG\_QUEUE, create record(s) on DEAL\_SKU\_TEMP for the RECLASS\_COST\_CHG\_QUEUE event. For reclassification events no location is given; therefore, these events are blown out to all item-location locations before inserting to DEAL\_SKU\_TEMP.

2. If an unprocessed record exists on RECLASS\_COST\_CHG\_QUEUE, create record(s) on DEAL\_SKU\_TEMP for any events on the FUTURE\_COST table that the RECLASS\_COST\_CHG\_QUEUE event affects (make the match using item, supplier, origin country id, location, and start date). For reclassification events (not cost change), no location is given and therefore these events are blown out to all item-location locations before seeking FUTURE\_COST matches.
3. If a record exists on DEAL\_SKU\_TEMP that affects one or more FUTURE\_COST records insert for those records on FUTURE\_COST into DEAL\_SKU\_TEMP to make sure they are recalculated.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                 |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 2 (daily)                                                                                                             |
| Scheduling Considerations | Must be run after DITINSRT and before COSTCALC in the deals batch cycle.                                                    |
| Pre-Processing            | Prepost precostcalc pre                                                                                                     |
| Post-Processing           | COSTCALC.PC                                                                                                                 |
| Threading Scheme          | Multiple processes can be run at the same time against different suppliers specified as the input parameter to the program. |

### Restart/Recovery

The LUW of this module is a single record from any one of the driving cursors, which all pick up item/supplier/origin country/location/active date combinations from RECLASS\_COST\_CHG\_QUEUE and FUTURE\_COST. The module has a restart/recovery based on item/ supplier/ origin country/ location/ active date and driving cursor number.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table                        | Select | Insert | Update | Delete |
|------------------------------|--------|--------|--------|--------|
| COST_CHANGE_ACTIVE_DATE_TEMP | Yes    | No     | No     | Yes    |
| RECLASS_COST_CHG_QUEUE       | Yes    | No     | Yes    | Yes    |
| ITEM_MASTER                  | Yes    | No     | No     | No     |
| DEPS                         | Yes    | No     | No     | No     |
| GROUPS                       | Yes    | No     | No     | No     |
| ITEM_LOC                     | Yes    | No     | No     | No     |
| DEAL_SKU_TEMP                | Yes    | Yes    | No     | No     |
| STORE                        | Yes    | No     | No     | No     |
| DISTRICT                     | Yes    | No     | No     | No     |
| REGION                       | Yes    | No     | No     | No     |
| AREA                         | Yes    | No     | No     | No     |
| FUTURE_COST                  | Yes    | No     | No     | Yes    |
| ITEM_LOC_TEMP                | Yes    | No     | No     | No     |

**I/O Specification**

NA

**vendinvc (Vendor Deal Invoicing)****Functional Area**

Deals Maintenance.

**Module Affected**

VENDINVC.PC

**Design Overview**

The batch module creates records in invoice match staging tables dealing for complex type deals.

VENDINVC.PC runs daily. It posts invoice information when the estimated next invoice date is reached and income is generated for a reporting period. In addition, it posts income for complex deals. It fills in the last invoice date with the end date of the last reporting period that is processed.

The invoicing logic is driven from the billing period estimated next invoice date for complex deals. The amount to be invoiced is the sum of the income accruals of the deal since the previous invoice date (or the deal start date for the first collection).

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                            |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3                                                                                                                                |
| Scheduling Considerations | Must be run before salmnth.pc, after dealact.pc and before the new programs, which perform forecast processing and DAILY_DATA roll up. |
| Pre-Processing            | prepost vendinvc pre - truncates STAGE_COMPLEX_DEAL_HEAD and STAGE_COMPLEX_DEAL_DETAIL tables                                          |
| Post-Processing           | prepost vendinvc post - calls the process_deal_head() function to update est_next_invoice_date of the deal to NULL.                    |
| Threading Scheme          | Threaded by deal id                                                                                                                    |

**Restart/Recovery**

The Logical Unit of Work for the program is a transaction consisting of deal\_id, deal\_detail\_id. When the max commit point is reached, the data is updated.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                     | Select | Insert | Update | Delete |
|---------------------------|--------|--------|--------|--------|
| DEAL_HEAD                 | Yes    | No     | Yes    | No     |
| DEAL_ACTUALS_ITEM_LOC     | Yes    | No     | No     | No     |
| DEAL_ACTUALS_FORECAST     | Yes    | No     | No     | No     |
| VAT_ITEM                  | Yes    | No     | No     | No     |
| STORE                     | Yes    | No     | No     | No     |
| WH                        | Yes    | No     | No     | No     |
| STAGE_COMPLEX_DEAL_HEAD   | No     | Yes    | No     | No     |
| STAGE_COMPLEX_DEAL_DETAIL | No     | Yes    | No     | No     |
| VENDINVC_TEMP             | Yes    | No     | No     | No     |
| PERIOD                    | Yes    | No     | No     | No     |

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS   | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**vendinvf (Vendor Deal Invoicing)****Functional Area**

Deals Maintenance.

**Module Affected**

VENDINV.FPC

**Design Overview**

The batch module creates records in staging tables dealing for fixed type deals.

VENDINV.FPC runs daily. It posts invoice information when the estimated next invoice date is reached and income is generated for a reporting period. In addition, it posts income for fixed deals. It fills in the last invoice date with the end date of the last reporting period that is processed.

The invoicing logic is driven by the collection dates for fixed deals. The amount to be invoiced is retrieved directly from fixed deal tables for a given deal date.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                  |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3                                                                                                                      |
| Scheduling Considerations | Must be run before salmnth.pc and before the new programs, which perform forecast processing and DAILY_DATA roll up.         |
| Pre-Processing            | prepost vendinvf pre - truncates STAGE_FIXED_DEAL_HEAD and STAGE_FIXED_DEAL_DETAIL tables                                    |
| Post-Processing           | prepost vendinvf post – calls the process_fixed_deal function to update the status of the fixed deal claim to 'I' (inactive) |
| Threading Scheme          | Threaded by deal id                                                                                                          |

**Restart/Recovery**

The Logical Unit of Work for the program is a transaction consisting of deal\_id and a collection date (date that the fixed deal should be claimed from the supplier). Data is committed to the database once the number of transactions processed reaches or exceeds the max\_commit\_ctr.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| FIXED_DEAL              | Yes    | No     | No     | No     |
| FIXED_DEAL_DATES        | Yes    | No     | No     | No     |
| FIXED_DEAL_MERCH        | Yes    | No     | No     | No     |
| FIXED_DEAL_MERCH_LOC    | Yes    | No     | No     | No     |
| SUBCLASS                | Yes    | No     | No     | No     |
| STAGE_FIXED_DEAL_HEAD   | No     | Yes    | No     | No     |
| STAGE_FIXED_DEAL_DETAIL | No     | Yes    | No     | No     |
| PERIOD                  | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS          | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES        | Yes    | No     | No     | No     |

**I/O Specification**

N/A



## Diff Ratio Build Batch

### Overview

Diff ratios are used to assist in ordering distribution. The ratio of Diff IDs is based on sales history. Relative sales history is recorded by size differentiators, women's lingerie for example.

### Batch Design Summary

The following batch designs are included in this functional area:

- DFRTBLD.PC (Diff Ratio Build)

### dfrtbld (Diff Ratio Build)

#### Functional Area

Diff ratio build

#### Module Affected

DFRTBLD.PC

#### Design Overview

The DFRTBLD.PC program reads the parameter setup on the DIFF\_RATIO\_HEAD table for diff\_ratio\_ids that were set up online and create size ratios for all size combinations that exist for the department/class/subclass. These ratios are also set up at the subclass/store level. When extracting sales history data, use sales history types (Clearance, Promotional, Regular) that have a value of 'Y' from the DIFF\_RATIO\_HEAD table to calculate the ratios.

Only diff\_ratio\_head records that are due to be rebuilt, or are recently updated online (that is, have an update indicator set to 'Y') and are system generated (that is, have a system indicator set to 'Y') is processed in the nightly run.

Diff\_ratio\_detail records are written for all locations for each diff\_ratio\_head record that is rebuildable. These hold sales per diffs/total sales for the subclass. There are also diff\_ratio\_detail records written for each store that the diff\_ratio's department/ class/ subclass/ sizes exist. These hold sales per diffs per store/total sales for the subclass at the store.

When extracting sales history from the Oracle Retail history tables, if the beginning or the end day to grab history lies in the middle of the week, then modify that date to be the previous end of week date.

For performance reasons, this program creates a comma delimited output data file for sql loader to upload data to table DIFF\_RATIO\_DETAIL. The control script for the sql loader is dfrtbld.ctl.

## Scheduling Constraints

| Schedule Information      | Description                                                                           |
|---------------------------|---------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                                       |
| Scheduling Considerations | This program is likely to run after sales information is uploaded into Oracle Retail. |
| Pre-Processing            | N/A                                                                                   |
| Post-Processing           | The SQL*Loader control file dfrtbld.ctl to load the data from ouput file.             |
| Threading Scheme          | Threaded by department                                                                |

## Restart/Recovery

This program is setup for multithreading and restart/recovery. The Logical Unit of Work is at subclass level and is threaded by department using the view v\_restart\_dept. The commit\_max\_ctr field on the RESTART\_CONTROL table determines the number of transactions that are processed before committing to the database.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| PERIOD            | Yes    | No     | No     | No     |
| DIFF_RATIO_HEAD   | Yes    | No     | Yes    | No     |
| DIFF_RATIO_DETAIL | No     | No     | No     | Yes    |
| DIFF_GROUP_DETAIL | Yes    | No     | No     | No     |
| V_RESTART_DEPT    | Yes    | No     | No     | No     |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| ITEM_LOC_HIST     | Yes    | No     | No     | No     |

## I/O Specification

This batch creates a comma delimited output data file for sql loader to upload data to table DIFF\_RATIO\_DETAIL. The control script for the sql loader is DFRTBLD.CTL.

## Electronic Data Interchange (EDI) Batch

### Overview

RMS supplies Pro\*C programs that batch process supplier data sent or received through electronic data interchange (EDI).

### RMS files and EDI Translations

RMS' EDI programs either create an output file if the data is being transmitted to the supplier or accept an input file initiated by the supplier. In all cases, the file is translated by the customer's EDI translation software application. For instance, if the supplier transmits an item list, the customer inputs the actual EDI transmitted file into its translation software. The data is populated to a standard Oracle Retail flat file that the batch module expects to process. Conversely, whenever RMS downloads purchase order data to the supplier, the batch module outputs the data in, again, an Oracle Retail standard flat file format. The customer inputs this data to its EDI translation software that creates the EDI output that is transmitted to the supplier.

### Batch Design Summary

The following batch designs are included in this functional area:

---

**Note:** The batch program, EDIDLINV.PC, has a functional connection to this chapter. For the design, please see the chapter, "Invoice Matching Batch," in this volume of the Operations Guide.

---

- EDIDLADD.PC (Store and Warehouse Address Download)
- EDIDLCON.PC (EDI Contract Information Download)
- EDIDLORD.PC (The New and Changed PO Download)
- EDIDLPRD.PC (Sales and Stock Activity Report Download)
- EDIPRG.PC (EDI Purge)
- EDIUPACK.PC (EDI Order Acknowledgement Upload)
- EDIUPADD.PC (New and Changed Supplier Address Upload)
- EDIUPAVL.PC (Supplier Availability for Contract Upload)
- EDIUPCAT.PC (New and Changed Upload from Supplier)

### edidladd (EDI Location Address to Vendor Download)

#### Functional Area

EDI – Suppliers

#### Module Affected

EDIDLADD.PC

## Design Overview

The purpose of this module is to download addresses of stores and warehouses to vendors.

The output file format is a standard Oracle Retail file format that is translated into EDI format by the Gentran translator. Addresses are downloaded in two different scenarios. The program downloads changes made to store or warehouse addresses into a flat file. Further, if a SYSTEM\_OPTIONS table flag (addr\_catalog) is set to true (Y), the addresses of all stores and warehouses are downloaded into a different file to be sent to suppliers.

## Scheduling Constraints

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad Hoc      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

## Restart/Recovery

Because of the lack of volume and the flexibility requirements of EDI, the program uses Oracle Retail's standard restart/recovery only minimally. The driving query volume is limited to the volume of the store and warehouse tables. Further, the output files are created if they do not exist and are overwritten if they already exist. In the event of a fatal error it is, therefore, reasonable to expect retailers to simply restart the job from the beginning without recovery.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| STORE           | Yes    | No     | No     | No     |
| ADDR            | Yes    | No     | Yes    | No     |
| ADD_TYPE_MODULE | Yes    | No     | No     | No     |
| ADD_TYPE        | Yes    | No     | No     | No     |
| WH              | Yes    | No     | No     | No     |

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS | Yes    | No     | Yes    | No     |
| PERIOD         | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name             | Field Type | Default Value                | Description                               |
|-------------|------------------------|------------|------------------------------|-------------------------------------------|
| FHEAD       | File head descriptor   | Char(5)    | FHEAD                        | Describes file line type                  |
|             | Line id                | Number(10) | 0000000001                   | Sequential file line number               |
|             | Gentran ID             | Char(4)    | 'DLADD'                      | Identifies file source                    |
|             | Purpose code           | Char(2)    | 04 (change) or 05 (replace)  | Add/change location or replace whole list |
| TDETL       | File record descriptor | Char(5)    | TDETL                        | Identifies file record type               |
|             | Line id                | Char(10)   | increment                    | Line number of file                       |
|             | Transaction number     | Number(10) | Start at 1,increment         | Identifies Transaction                    |
|             | Date                   | Char(8)    |                              | Period.vdate YYYYMMDD                     |
|             | Store or warehouse     | Char(2)    | SN (store) or WH (warehouse) | Location type                             |
|             | Location               | Number(10) |                              | Store.store or wh.wh                      |
|             | Location name          | Char(150)  |                              | Store.store_name or wh.wh_name            |
|             | Address line 1         | Char(240)  |                              | addr.add_1                                |
|             | Address line 2         | Char(240)  |                              | addr.add_2                                |
|             | City                   | Char(120)  |                              | addr.city                                 |
|             | State                  | Char(3)    |                              | addr.state                                |
|             | Postal code            | Char(30)   |                              | addr.post                                 |
|             | Country                | Char(3)    |                              | addr.country_id                           |

| Record Name | Field Name               | Field Type | Default Value | Description                                                  |
|-------------|--------------------------|------------|---------------|--------------------------------------------------------------|
|             | Address Type Description | Char(40)   |               | add_type.type_desc                                           |
| FTAIL       | File record descriptor   | Char(5)    | FTAIL         | Marks end of file                                            |
|             | Total No Lines           | Char(10)   |               | Total lines in file                                          |
|             | Total No TDETL lines     | Number(10) |               | Total number of transaction lines excluding FHEAD and FTAIL. |

## edidlcon (EDI Contract Information Downloads)

### Functional Area

EDI – Contracts

### Module Affected

EDIDLCN.PC

### Design Overview

This program is used to download EDI contract information. Contracts are only processed if they are in approved status and have an edi\_contract\_ind of 'Y' on the CONTRACT\_HEADER table. The output file of this program contains all records for the supplier contract data which are in approved status.

### Scheduling Constraints

| Schedule Information      | Description                  |
|---------------------------|------------------------------|
| Processing Cycle          | Phase 4 (Daily) or as needed |
| Scheduling Considerations | N/A                          |
| Pre-Processing            | N/A                          |
| Post-Processing           | N/A                          |
| Threading Scheme          | N/A                          |

### Restart/Recovery

The logical unit of work for this program is set at the contract number. This program processes one contract number at a time.

### Locking Strategy

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| CONTRACT_HEADER | Yes    | No     | Yes    | No     |
| CONTRACT_COST   | Yes    | No     | No     | No     |
| ITEM_MASTER     | Yes    | No     | No     | No     |
| CONTRACT_DETAIL | Yes    | No     | No     | No     |
| WH              | Yes    | No     | No     | No     |
| ITEM_SUPPLIER   | Yes    | No     | No     | No     |
| DIFF_IDS        | Yes    | No     | No     | No     |

**I/O Specification****Output File Layout**

| Record Name | Field Name           | Field Type | Default Value | Description                                |
|-------------|----------------------|------------|---------------|--------------------------------------------|
| FHEAD       | File head descriptor | Char(5)    | FHEAD         | Describes file line type.                  |
|             | Line Number          | Number(10) | 0000000001    | Sequential file line number                |
|             | Gentran ID           | Char(4)    | 'DNCN'        | Identifies which translation Gentran uses. |
|             | Current date         | Char(14)   |               | Vdate in YYYYMMDDHH24MISS format           |
| THEAD       | File head descriptor | Char(5)    | THEAD         | Describes file line type.                  |
|             | Line Number          | Number(10) |               | Sequential file line number                |
|             | Transaction Number   | Number(10) |               | Sequential transaction number              |
|             | Supplier             | Number(10) |               | Supplier number                            |
|             | Contract Number      | Number(6)  |               | Contract number                            |
|             | Contract type        | Char(1)    |               | Type of contract                           |
|             | Department           | Number(4)  |               | Department                                 |

| Record Name | Field Name             | Field Type | Default Value | Description                                                                                                                             |
|-------------|------------------------|------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------|
|             | Currency code          | Char(3)    |               | Currency code                                                                                                                           |
|             | Total contract cost    | Number(20) |               | Total cost of contract * 10000 (4 implied decimal places)                                                                               |
| TDETL       | File record descriptor | Char(5)    | TDETL         | Describes file line type                                                                                                                |
|             | Line Number            | Number(10) |               | Sequential file line number                                                                                                             |
|             | Transaction number     | Number(10) |               | Sequential transaction number                                                                                                           |
|             | Item Number Type       | Char(6)    |               | Item type                                                                                                                               |
|             | Item Number            | Char(25)   |               | Item                                                                                                                                    |
|             | Ref Item Number Type   | Char(6)    |               | Reference item number type                                                                                                              |
|             | Ref Item Number        | Char(25)   |               | Primary reference item                                                                                                                  |
|             | Diff1                  | Char(120)  |               | Diff1 Description                                                                                                                       |
|             | Diff2                  | Char(120)  |               | Diff2 Description                                                                                                                       |
|             | Diff3                  | Char(120)  |               | Diff3 Description                                                                                                                       |
|             | Diff4                  | Char(120)  |               | Diff4 Description                                                                                                                       |
|             | VPN                    | Char(30)   |               | Vendor Product Number for an item                                                                                                       |
|             | Unit cost              | Number(20) |               | Unit cost of item *10000 (4 implied decimal places)                                                                                     |
|             | Ready Date             | Char(14)   |               | Date on which the items are provided by supplier.<br>This field contains only values for contract types of 'A' or 'B'.                  |
|             | Ready Quantity         | Number(20) |               | Quantity contracted with supplier*10000 (4 implied decimal points)<br>This field contains only values for contract types of 'A' or 'B'. |



| Record Name | Field Name             | Field Type | Default Value | Description                                                                                           |
|-------------|------------------------|------------|---------------|-------------------------------------------------------------------------------------------------------|
|             | Location Type          | Char(2)    |               | 'ST' (store) or 'WH' (warehouse)<br>This field contains only values for contract types of 'A' or 'B'. |
|             | Location number        | Number(10) |               | Location number<br>This field contains only values for contract types of 'A' or 'B'.                  |
| TTAIL       | File Record descriptor | Char(5)    | TTAIL         | Describes file line type                                                                              |
|             | Line Number            | Number(10) |               | Sequential file line number                                                                           |
|             | Transaction number     | Number(10) |               | Sequential transaction number                                                                         |
| FTAIL       | File record descriptor | Char(5)    | FTAIL         | Marks the end of file                                                                                 |
|             | Line number            | Number(10) |               | Sequential file line number                                                                           |
|             | Number of lines        | Number(10) |               | Number of lines in file not counting FHEAD and FTAIL                                                  |

## edidlord (EDI purchase order download)

### Functional Area

EDI – Purchase Orders

### Module Affected

EDIDLORD.PC

### Design Overview

Orders created within the Oracle Retail system are written to a flat file if they are approved and marked as EDI orders. This module is used to write new and changed purchase order data to a flat file in the Oracle Retail standard format. The translation to EDI format takes place via an outside translator such as Gentran. The order revision tables and allocation revision tables are also used to ensure that the latest changes are being sent and to allow both original and modified values to be sent. These revision tables are populated during the online ordering process and the batch replenishment process whenever an order is approved, and constitutes a history of all revisions to the order.

If multi-channel is turned ON in the system, the program sums up all quantities to the physical warehouse level for an order, before writing it into the output file.

If shipments are to be pre-marked by the supplier for cross docking, then along with the order information: allocation, location and quantities are also sent.

If the backhaul type is specified as “Calculated”, then the backhaul allowances are calculated.

If the order contains pack items; hierarchical pack information is sent (this may include outer packs, inner packs, and fashion styles with associated pack templates as well as component item information).

### Scheduling Constraints

| Schedule Information      | Description                                                           |
|---------------------------|-----------------------------------------------------------------------|
| Processing Cycle          | Phase 4 (Daily), can also be run ad-hoc multiple times per day        |
| Scheduling Considerations | This program needs to be scheduled after replenishment and ORDREV.PC. |
| Pre-Processing            | N/A                                                                   |
| Post-Processing           | N/A                                                                   |
| Threading Scheme          | Multi-threaded by supplier                                            |

### Restart/Recovery

The logical unit of work for this program is set at the supplier level. Threading is performed by the supplier using the v\_restart\_supplier view.

Restart ability is implied because the program updates ordhead.edi\_sent\_ind as records and are written out. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O. The recommended commit counter setting is 10000 records.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table       | Select | Insert | Update | Delete |
|-------------|--------|--------|--------|--------|
| ORDHEAD     | Yes    | No     | Yes    | No     |
| ORDHEAD_REV | Yes    | No     | No     | No     |
| TERM        | Yes    | No     | No     | No     |
| SUPS        | Yes    | No     | No     | No     |
| ORDSKU      | Yes    | No     | No     | No     |
| ORDSKU_REV  | Yes    | No     | No     | No     |

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| ITEM_MASTER           | Yes    | No     | No     | No     |
| ORDLOC                | Yes    | No     | No     | No     |
| ORDLOC_REV            | Yes    | No     | No     | No     |
| ORDLOC_DISCOUNT       | Yes    | No     | No     | No     |
| ALLOC_HEADER          | Yes    | No     | No     | No     |
| ALLOC_DETAIL          | Yes    | No     | No     | No     |
| ALLOC_REV             | Yes    | No     | No     | No     |
| WH                    | Yes    | No     | No     | No     |
| PACKITEM_BREAKOUT     | Yes    | No     | No     | No     |
| SUPS_PACK_TMPL_DESC   | Yes    | No     | No     | No     |
| ITEM_SUPPLIER         | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_DIM | Yes    | No     | No     | No     |
| STORE                 | Yes    | No     | No     | No     |
| ADDR                  | Yes    | No     | No     | No     |

## I/O Specification

### Output File

For a new order, the “old” fields should be blank. For a changed order, both old and new fields should hold values. If the value has changed. “Old” values come from the revision tables for the latest revision before the current one (the last one sent), while new orders come from the ordering tables.

FHEAD – REQUIRED: File identification, one line per file.

TORDR – REQUIRED: Order level information, one line per order.

TITEM – REQUIRED: Item description, multiple lines per order possible.

TPACK – OPTIONAL: Pack contents, multiple lines per order possible. This line is written only for pack items.

TSHIP – REQUIRED: Ship to location and quantity, allocation location, multiple lines per item possible. Allocation information is optional on this line—exists if premark\_ind is ‘Y’.

TTAIL – REQUIRED: Order end, one line per order.

FTAIL – REQUIRED: End of file marker, one line per file.

| Record Name | Field Name        | Field Type | Default Value | Description      |
|-------------|-------------------|------------|---------------|------------------|
| FHEAD       | Record descriptor | Char(5)    | FHEAD         | File head marker |
|             | Line id           | Number(10) | 0000000001    | Unique line id   |

| Record Name | Field Name                                    | Field Type | Default Value | Description                                             |
|-------------|-----------------------------------------------|------------|---------------|---------------------------------------------------------|
|             | Translator id                                 | Char(5)    | DLORD         | Identifies transaction type                             |
|             | File create date                              | Char(14)   |               | Vdate in YYYYMMDDHH24MISS format                        |
| TORDR       | Record descriptor                             | Char(5)    | TORDR         | Order header information                                |
|             | Line id                                       | Number(10) |               | Unique file line id                                     |
|             | Transaction id                                | Number(10) |               | Unique transaction id                                   |
|             | Order change type                             | Char(2)    |               | 'CH' (changed) or 'NW' (new)                            |
|             | Order number                                  | Number(8)  |               | Internal Oracle Retail order no                         |
|             | Supplier                                      | Number(10) |               | Internal Oracle Retail supplier id                      |
|             | Vendor order id                               | Char(15)   |               | External vendor_order_no (if available)                 |
|             | Order written date                            | Char(14)   |               | Order created date in YYYYMMDDHH24MISS format           |
|             | Original order approval date                  | Char(14)   |               | Original order approval date in YYYYMMDDHH24MISS format |
|             | Old Currency Code                             | Char(3)    |               | Old order currency_code (ISO standard)                  |
|             | New Currency Code                             | Char(3)    |               | Changed order currency_code (ISO standard)              |
|             | Old Shipment Method of payment                | Char(2)    |               | Old ship_pay_method                                     |
|             | New Shipment Method of Payment                | Char(2)    |               | Changed ship_pay_method                                 |
|             | Old Transportation Responsibility             | Char(2)    |               | Old fob_trans_res                                       |
|             | Old Transportation Responsibility Description | Char(250)  |               | Old fob_trans_res_desc                                  |

| Record Name | Field Name                        | Field Type | Default Value | Description                                        |
|-------------|-----------------------------------|------------|---------------|----------------------------------------------------|
|             | New Transportation Responsibility | Char(2)    |               | Changed fob_trans_res                              |
|             | New Trans. Resp. Description      | Char(250)  |               | New fob_trans_res_desc                             |
|             | Old Title Passage Location        | Char(2)    |               | Old fob_title_pass                                 |
|             | New Title Passage Location        | Char(2)    |               | Changed fob_title_pass                             |
|             | Old Title Passage Description     | Char(250)  |               | Old fob_title_pass_desc                            |
|             | New Title Passage Description     | Char(250)  |               | Changed fob_title_pass_desc                        |
|             | Old not before date               | Char(14)   |               | Old not_before_date in YYYYMMDDHH24MISS format     |
|             | New not before date               | Char(14)   |               | Changed not_before_date in YYYYMMDDHH24MISS format |
|             | Old not after date                | Char(14)   |               | Old not_after_date in YYYYMMDDHH24MISS format      |
|             | New not after date                | Char(14)   |               | Changed not_after_date in YYYYMMDDHH24MISS format  |
|             | Old Purchase type                 | Char(6)    |               | Old Purchase type                                  |
|             | New Purchase type                 | Char(6)    |               | New Purchase type                                  |
|             | Backhaul allowance                | Char(20)   |               | Backhaul allowance                                 |
|             | Old terms description             | Char(240)  |               | Old terms description from terms table             |
|             | New terms description             | Char(240)  |               | New terms description from terms table             |
|             | Old pickup date                   | Char(14)   |               | Old pickup date<br>YYYYMMDDHH24MISS                |
|             | New pickup date                   | Char(14)   |               | New pickup date<br>YYYYMMDDHH24MISS                |

| Record Name | Field Name               | Field Type | Default Value | Description                                                   |
|-------------|--------------------------|------------|---------------|---------------------------------------------------------------|
|             | Old ship method          | Char(6)    |               | Old ship method                                               |
|             | New ship method          | Char(6)    |               | New ship method                                               |
|             | Old comment description  | Char(2000) |               | Old comment description                                       |
|             | New comment description  | Char(2000) |               | New comment description                                       |
|             | Supplier DUNS number     | Char(9)    |               | Supplier DUNS number                                          |
|             | Supplier DUNS location   | Char(4)    |               | Supplier DUNS location                                        |
| TITEM       | File record descriptor   | Char(5)    | TITEM         | Item info                                                     |
|             | Line id                  | Number(10) |               | Unique line id                                                |
|             | Transaction id           | Number(10) |               | Unique transaction id                                         |
|             | Item Number Type         | Char(6)    |               | Item_number_type                                              |
|             | Item                     | Char(25)   |               | Item (For a pack item, this is the pack number)               |
|             | Old Ref Item Number type | Char(6)    |               | Item_number_type for old ref_item                             |
|             | Old Ref Item             | Char(25)   |               | Old Ref_Item                                                  |
|             | New Ref Item Number type | Char(6)    |               | Item_number_type for new ref_item                             |
|             | New Ref Item             | Char(25)   |               | Changed Ref_Item                                              |
|             | Vendor catalog number    | Char(30)   |               | Supplier_item (VPN)                                           |
|             | Free Form Description    | Char(250)  |               | Item_desc                                                     |
|             | Supplier Diff 1          | Char(120)  |               | Supplier's diff 1                                             |
|             | Supplier Diff 2          | Char(120)  |               | Supplier's diff 2                                             |
|             | Supplier Diff 3          | Char(120)  |               | Supplier's diff 3                                             |
|             | Supplier Diff 4          | Char(120)  |               | Supplier's diff 4                                             |
|             | Pack Size                | Number(12) |               | Supplier defined pack size * 10000 (4 implied decimal places) |

| Record Name | Field Name                | Field Type | Default Value | Description                                                           |
|-------------|---------------------------|------------|---------------|-----------------------------------------------------------------------|
| TPACK       | File record descriptor    | Char(5)    | TPACK         | Pack component info                                                   |
|             | Line id                   | Number(10) |               | Unique line id                                                        |
|             | Transaction id            | Number(10) |               | Unique transaction id                                                 |
|             | Pack id                   | Char(25)   |               | Packitem_breakout.pack_no (same as item for the pack item)            |
|             | Inner pack id             | Char(25)   |               | Inner pack identification                                             |
|             | Pack Quantity             | Number(12) |               | Packitem_breakout.pack_item_qty*10000 (4 implied decimal places)      |
|             | Component Pack Quantity   | Number(12) |               | Packitem_breakout.comp_pack_qty*10000 (4 implied decimal places)      |
|             | Item Parent Part Quantity | Number(12) |               | Packitem_breakout.item_parent_pt_qty*10000 (4 implied decimal places) |
|             | Item Quantity             | Number(12) |               | Packitem_breakout.item_qty*10000 (4 implied decimal places)           |
|             | Item Number Type          | Char(6)    |               | Item number type                                                      |
|             | Item                      | Char(25)   |               | Item                                                                  |
|             | Ref Item Number Type      | Char(6)    |               | Ref_item_number_type                                                  |
|             | Ref Item                  | Char(25)   |               | Ref_item                                                              |
|             | VPN                       | Char(30)   |               | Supplier item (vpn)                                                   |
|             | Supplier Diff 1           | Char(120)  |               | Supplier's diff 1                                                     |
|             | Supplier Diff 2           | Char(120)  |               | Supplier's diff 2                                                     |
|             | Supplier Diff 3           | Char(120)  |               | Supplier's diff 3                                                     |
|             | Supplier Diff 4           | Char(120)  |               | Supplier's diff 4                                                     |
|             | Item Parent               | Char(25)   |               | Required when Pack Template is not NULL                               |
|             | Pack template             | Number(8)  |               | Pack template associated w/style (packitem_breakout.pack_tmpl_id)     |

| Record Name | Field Name               | Field Type | Default Value | Description                                                                                                                |
|-------------|--------------------------|------------|---------------|----------------------------------------------------------------------------------------------------------------------------|
|             | Template description     | Char(250)  |               | Description of pack template.<br>supp_pack_tmpl_desc.supp_pack_desc                                                        |
| TSHIP       | Record type              | Char(5)    | TSHIP         | Describes the file record-shipment information                                                                             |
|             | Line id                  | Number(10) |               | Unique file line number                                                                                                    |
|             | Transaction id           | Number(10) |               | Unique transaction number                                                                                                  |
|             | Location type            | Char(2)    |               | 'ST' store or 'WH' warehouse                                                                                               |
|             | Ship to location         | Number(10) |               | Location value form ordloc (store or warehouse)                                                                            |
|             | Old unit cost            | Number(20) |               | Old unit cost*10000 (4 implied decimal places)                                                                             |
|             | New unit cost            | Number(20) |               | New unit cost*10000 (4 implied decimal places)                                                                             |
|             | Old quantity             | Number(12) |               | Old qty_ordered *10000 or qty_allocated*10000 (4 implied decimal places)                                                   |
|             | New quantity             | Number(12) |               | Changed qty_ordered*10000 or qty_allocated*10000 (4 implied decimal places)                                                |
|             | Old outstanding quantity | Number(12) |               | Old (qty_ordered-qty_received)*10000 or (qty_allocated-qty_transferred)*10000 for an allocation (4 implied decimal places) |
|             | New outstanding quantity | Number(12) |               | Changed qty_ordered-qty_received (4 implied decimal places)(or qty_allocated-qty_transferred, for an allocation)           |
|             | Cancel code              | Char(1)    |               |                                                                                                                            |
|             | Old cancelled quantity   | Number(12) |               | Previous quantity cancelled (4 implied decimal places)                                                                     |
|             | New cancelled quantity   | Number(12) |               | Changed quantity cancelled (4 implied decimal places)                                                                      |
|             | Quantity type flag       | Char(1)    |               | 'S'hip to 'A'llocate                                                                                                       |



| Record Name | Field Name                         | Field Type | Default Value | Description                                   |
|-------------|------------------------------------|------------|---------------|-----------------------------------------------|
|             | Store or warehouse indicator       | Char(2)    |               | 'ST' (store) or 'WH' (warehouse)              |
|             | Old x-dock location                | Number(10) |               | Alloc_detail location (store or wh)           |
|             | New x-dock location                | Number(10) |               | Alloc_detail location (store or wh)           |
|             | Case length                        | Number(12) |               | Case length (4 implied decimal places)        |
|             | Case width                         | Number(12) |               | Case width (4 implied decimal places)         |
|             | Case height                        | Number(12) |               | Case height (4 implied decimal places)        |
|             | Case LWH unit of measure           | Char(4)    |               | Case LWH unit of measure                      |
|             | Case weight                        | Number(12) |               | Case weight (4 implied decimal places)        |
|             | Case weight unit of measure        | Char(4)    |               | Case weight unit of measure                   |
|             | Case liquid volume                 | Number(12) |               | Case liquid volume (4 implied decimal places) |
|             | Case liquid volume unit of measure | Char(4)    |               | Case liquid volume unit of measure            |
|             | Location DUNS number               | Char(9)    |               | Location DUNS number                          |
|             | Location DUNS loc                  | Char(4)    |               | Location DUNS loc                             |
|             | Old unit cost init                 | Number(20) |               | Old unit cost init (4 implied decimal places) |
|             | New unit cost init                 | Number(20) |               | New unit cost init (4 implied decimal places) |
|             | Item/loc discounts                 | Number(20) |               | Item/loc discounts (4 implied decimal places) |
| TTAIL       | Record type                        | Char(5)    | TTAIL         | Describes file record – marks end of order    |
|             | Line id                            | Number(10) |               | Unique file line id                           |
|             | Transaction id                     | Number(10) |               | Unique transaction id                         |

| Record Name | Field Name            | Field Type | Default Value | Description                                                               |
|-------------|-----------------------|------------|---------------|---------------------------------------------------------------------------|
|             | #Lines in transaction | Number(10) |               | Number of lines in transaction                                            |
| FTAIL       | Record type           | Char(5)    | FTAIL         | Describes file record – marks end of file                                 |
|             | Line id               | Number(10) |               | Unique file line id                                                       |
|             | #lines                | Number(10) |               | Total number of transaction lines in file (not including FHEAD and FTAIL) |

## edidlprd (EDI Sales and Stock On Hand Report Download)

### Functional Area

EDI – Sales and Inventory

### Module Affected

EDIDLPRD.PC

### Design Overview

The following guideline describes Oracle Retail's implementation of the ANSI 852 Product Activity Data transaction set.

The productivity data report is a sales audit summary that is sent to specified EDI vendors, giving sales details, current stock on hand for all location, and current in transit quantities for each of the items primarily supplied by that vendor.

This program polls all suppliers that require activity reporting. Those that require daily reports have daily sales information selected from the EDI\_DAILY\_SALES table, while those requiring weekly reports pull information from the ITEM\_LOC\_HIST and ITEM\_MASTER tables to get the weekly sales data. The store level table ITEM\_LOC is queried for current stock levels, and the quantities that are in-transit are derived from the transfer tables.

### Scheduling Constraints

| Schedule Information      | Description                                                                                      |
|---------------------------|--------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 4 (Daily), or as needed.                                                                   |
| Scheduling Considerations | N/A                                                                                              |
| Pre-Processing            | Pre-processing queries and inserts all candidate suppliers to EDI_SUPS_TEMP table.               |
| Post-Processing           | Post-processing deletes data from EDI_DAILY_SALES table.                                         |
| Threading Scheme          | Multi-threaded by supplier through the locking of EDI_SUPS_TEMP table for each supplier fetched. |

## Restart/Recovery

Restart/recovery in this program is achieved through utilizing the global temporary table EDI\_SUPS\_TEMP. Once a supplier is processed, it is deleted from the EDI\_SUPS\_TEMP table to prevent the same supplier from being processed again during recovery.

## Locking Strategy

This module locks the edi\_sups\_temp supplier record to prevent multiple threads from processing the same supplier. Once a supplier is processed, it is deleted from EDI\_SUPS\_TEMP table.

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| SUPS                  | Yes    | No     | No     | No     |
| EDI_SUPS_TEMP         | Yes    | No     | No     | Yes    |
| EDI_DAILY_SALES       | Yes    | Yes    | Yes    | No     |
| PERIOD                | Yes    | No     | No     | No     |
| COMPHEAD              | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS        | Yes    | No     | No     | No     |
| STORE                 | Yes    | No     | No     | No     |
| WH                    | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| ITEM_SUPPLIER         | Yes    | No     | No     | No     |
| ITEM_LOC_HIST         | Yes    | No     | No     | No     |
| ITEM_LOC_SOH          | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name             | Field Type | Default Value | Description                                                                            |
|-------------|------------------------|------------|---------------|----------------------------------------------------------------------------------------|
| FHEAD       | File record descriptor | Char(5)    | FHEAD         | Describes record type                                                                  |
|             | Line number            | Number(10) | 0000000001    | Sequential file line number                                                            |
|             | File source            | Char(5)    | DLPRD         | File Type                                                                              |
|             | File create date       | Char(8)    | Period.vdate  | Vdate in YYYYMMDD format                                                               |
| THEAD       | File record descriptor | Char(5)    | THEAD         | Identifies record type                                                                 |
|             | Line number            | Number(10) |               | Sequential file line number                                                            |
|             | Transaction number     | Number(10) |               | Sequential transaction number                                                          |
|             | Report date            | Char(8)    |               | Vdate-lag days between reporting (vdate if supplier reports weekly) in YYYYMMDD format |
|             | Supplier               | Number(10) |               | Supplier Number                                                                        |
| TITEM       | File record descriptor | Char(5)    | TITEM         | Identifies file record type                                                            |
|             | Line number            | Number(10) |               | Sequential file line number                                                            |
|             | Transaction number     | Number(10) |               | Sequential transaction number                                                          |
|             | Item                   | Char(25)   |               | Item                                                                                   |
|             | Item_Num_Type          | Char(6)    |               | Item Number Type                                                                       |
|             | Ref_Item               | Char(25)   |               | Reference item                                                                         |
|             | Ref_Item_Num_Type      | Char(6)    |               | Reference Item Number Type                                                             |
|             | Vendor catalog number  | Char(30)   |               | VPN (Vendor Product Number)                                                            |
|             | Item description       | Char(250)  |               | Item description (item desc)                                                           |
| TQUTY       | File record descriptor | Char(5)    | TQUTY         | Identifies record type                                                                 |
|             | Line number            | Number(10) |               | Sequential file line number                                                            |

| Record Name | Field Name                  | Field Type | Default Value | Description                                                                                  |
|-------------|-----------------------------|------------|---------------|----------------------------------------------------------------------------------------------|
|             | Transaction number          | Number(10) |               | Sequential transaction number                                                                |
|             | Quantity descriptor         | Char(15)   |               | 'On-hand' (stock)/'Sold'(sales)/'In transit'                                                 |
|             | Location type               | Char(2)    |               | 'ST' for store or 'WH' warehouse                                                             |
|             | Location                    | Number(10) |               | Store or warehouse number                                                                    |
|             | Unit cost                   | Number(20) |               | Unit cost (4 implied decimal places) from item_supp_country_loc table (in supplier currency) |
|             | Quantity                    | Number(12) |               | Quantity – 4 implied decimal places                                                          |
| TTAIL       | File record descriptor      | Char(5)    | TTAIL         | Identifies record type                                                                       |
|             | Line number                 | Number(10) |               | Sequential file line number                                                                  |
|             | Transaction lines           | Number(6)  |               | Number of lines for this transaction                                                         |
| FTAIL       | File record descriptor      | Char(5)    |               | Identifies record type                                                                       |
|             | Line number                 | Number(10) |               | Total number of lines in file.                                                               |
|             | Number of transaction lines | Number(10) |               | Number of transaction lines in file.                                                         |

## ediprg (EDI Purge)

### Functional Area

EDI

### Module Affected

EDIPRG.PC

### Design Overview

This program purges rejected EDI “new items” or cost changes on the edi temporary tables. New or changed item information as well as cost change information sent by suppliers are approved or rejected on-line. The items that are rejected are removed from the system when they exist for longer than the number of days specified on the SYSTEM\_VARIABLES table. From SYSTEM\_OPTIONS table the two columns edi\_new\_item\_days and edi\_cost\_chg\_days are used for EDI purging.

### Scheduling Constraints

| Schedule Information      | Description                         |
|---------------------------|-------------------------------------|
| Processing Cycle          | Ad Hoc (Monthly)                    |
| Scheduling Considerations | Towards the end of the batch cycle. |
| Pre-Processing            | N/A                                 |
| Post-Processing           | N/A                                 |
| Theading Scheme           | N/A                                 |

### Restart/Recovery

This program uses the commit\_max\_ctr on the RESTART\_CONTROL table to periodically commit SQL delete operations.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table        | Select | Insert | Update | Delete |
|--------------|--------|--------|--------|--------|
| EDI_COST_CHG | Yes    | No     | No     | Yes    |
| EDI_NEW_ITEM | No     | No     | No     | Yes    |
| EDI_COST_LOC | No     | No     | No     | Yes    |

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |

**I/O Specification**

N/A

## ediupack (EDI Supplier Order Acknowledgements and Changes)

**Functional Area**

EDI – Purchase Orders

**Module Affected**

EDIUPACK.PC

**Design Overview**

This program has four functions: to acknowledge vendor receipt of a buyer-generated order without changes, to acknowledge vendor receipt of a buyer-generated order with date, cost or quantity modifications, to notify buyer of a vendor-generated order, and to acknowledge order cancellations.

All acknowledgements update the ORDHEAD table with acknowledgement information.

When the supplier sends the acknowledgement with modifications, it can send the entire purchase order or only the changes. The file details are matched to the current order. If the Not Before Date, Not After Date, Quantity, Price, and item all match the current order, then no changes were submitted. If one of the variables is blank, for example the price, assume that no pricing changes were made. As soon as one of the variables does not match, the order is changed. These changes are not written directly to the order; they are written to the revision tables. Revisions are accepted in the ordering dialog and changed orders are resubmitted via EDIDLORD.

Vendor generated orders creates new orders by inserting new records on the EDI temporary order tables.

For more information about supplier contracts, see the chapter “Contracts Batch” in this volume of the RMS Operations Guide.

**Scheduling Constraints**

| Schedule Information      | Description     |
|---------------------------|-----------------|
| Processing Cycle          | Phase 2 (Daily) |
| Scheduling Considerations | N/A             |
| Pre-Processing            | N/A             |
| Post-Processing           | N/A             |
| Threading Scheme          | N/A             |

### Restart/Recovery

The files do not have enough volume to warrant the implementation of restart recovery for commit/rollback considerations, but minimal file-based restart/recovery capability is added. The logical unit of work is a complete transaction represented by detail lines between the transaction header and transaction tail.

A savepoint is issued before each transaction header record is successfully processed. If a non-fatal error occurs, a rollback to the last savepoint is issued so that the rejected records are not posted to the database. If a fatal error occurs and restart is necessary, processing restarts at the last commit point.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| PERIOD                | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS        | Yes    | No     | No     | No     |
| EDI_ORD_TEMP          | No     | Yes    | Yes    | No     |
| DAILY_PURGE           | Yes    | No     | No     | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| ITEM_LOC              | Yes    | No     | No     | No     |
| ITEM_SUPPLIER         | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | Yes    | Yes    | Yes    | No     |
| ORDHEAD               | Yes    | No     | Yes    | No     |
| ORDLOC                | Yes    | No     | No     | No     |
| ORDSKU                | Yes    | No     | No     | No     |
| ORDHEAD_REV           | Yes    | Yes    | No     | No     |
| ORDLOC_REV            | No     | Yes    | Yes    | No     |
| ORDSKU_REV            | No     | Yes    | No     | No     |
| PRICE_HIST            | No     | Yes    | No     | No     |
| POS_MODS              | No     | Yes    | No     | No     |
| ITEM_LOC_SOH          | No     | Yes    | No     | No     |



## I/O Specification

### Input File Layout

| Record Name | Field Name             | Field Type | Default Value       | Description                                                                          |
|-------------|------------------------|------------|---------------------|--------------------------------------------------------------------------------------|
| FHEAD       | File head descriptor   | Char(5)    | FHEAD               | Describes file line type.                                                            |
|             | Line id                | Number(10) | 0000000001          | Sequential file line number                                                          |
|             | File Type Definition   | Char(4)    | ORAK                | Identifies file as 'Order Acknowledgment Import'.                                    |
| THEAD       | File record descriptor | Char(5)    | THEAD               | Describes file line type                                                             |
|             | Line id                | Number(10) | Line number in file | Sequential file line number                                                          |
|             | Transaction number     | Number(10) |                     | Sequential transaction number                                                        |
|             | Acknowledge type       | Char(2)    |                     | AP-product replenishment<br>AK- Acknowledge or change<br>CA-cancel order (no detail) |
|             | Order number           | Char(15)   |                     | May be external order number (vendor order number) OR Oracle Retail order number     |
|             | Written_date           | Char(8)    |                     | Written date in YYYYMMDD format                                                      |
|             | Supplier number        | Number(10) |                     | Supplier number                                                                      |
|             | Not before date        | Char(8)    |                     | Not_before_date<br>YYYYMMDD                                                          |
|             | Not after date         | Char(8)    |                     | Not_after_date<br>YYYYMMDD                                                           |
|             | Purchase type          | Char(6)    |                     | Specifies type of purchase – may be blank                                            |
|             | Pickup date            | Char(8)    |                     | Pickup_date YYYYMMDD – may be blank                                                  |
|             |                        |            |                     |                                                                                      |
| TITEM       | File record descriptor | Char(5)    | TITEM               | Describes file line type                                                             |
|             | Line id                | Number(10) | Line number in file | Sequential file line number                                                          |

| Record Name | Field Name             | Field Type | Default Value       | Description                                              |
|-------------|------------------------|------------|---------------------|----------------------------------------------------------|
|             | Transaction number     | Number(10) |                     | Sequential transaction number                            |
|             | ITEM                   | Char(25)   |                     | Item (either item or ref_item must be defined)           |
|             | Ref_item               | Char(25)   |                     | Reference item (either item or ref_item must be defined) |
|             | Vendor catalog number  | Char(30)   |                     | VPN (Vendor Product Number)                              |
|             | Unit cost value        | Number(20) |                     | Unit_cost * 10000 (4 implied decimal places)             |
|             | Loc_type               | Char(2)    |                     | 'ST' for store, 'WH' for warehouse                       |
|             | Location               | Number(10) |                     | If NULL, apply to all locations for this item.           |
|             | Pickup location        | Char(250)  |                     | Location to pick up item – may be blank                  |
| TSHIP       | File record descriptor | Char(5)    | TSHIP               | Describes file line type                                 |
|             | Line id                | Number(10) | Line number in file | Sequential file line number                              |
|             | Transaction number     | Number(10) |                     | Sequential transaction number                            |
|             | Store/wh indicator     | Char(2)    |                     | 'ST' for store, 'WH' for warehouse                       |
|             | Ship to location       | Number(10) |                     | Store or warehouse number                                |
|             | Quantity               | Number(12) |                     | Quantity ordered * 10000 (4 implied decimal places)      |
| TTAIL       | File record descriptor | Char(5)    | TTAIL               | Describes file line type                                 |
|             | Line id                | Number(10) | Line number in file | Sequential file line number                              |
|             | Transaction number     | Number(10) |                     | Sequential transaction number                            |
|             | Lines in transaction   | Number(6)  |                     | Total number of lines in this transaction                |
| FTAIL       | File record descriptor | Char(5)    | FTAIL               | Marks end of file                                        |

| Record Name | Field Name             | Field Type | Default Value       | Description                             |
|-------------|------------------------|------------|---------------------|-----------------------------------------|
|             | Line id                | Number(10) | Line number in file | Sequential file line number             |
|             | Number of transactions | Number(10) |                     | Number of lines between FHEAD and FTAIL |

## ediupadd (EDI Supplier Address Upload)

### Functional Area

EDI – Suppliers

### Module Affected

EDIUPADD.PC

### Design Overview

The EDIUPADD.PC batch program is used to read vendor/supplier sent EDI 838 Profile Data Files. These files are processed by vendor/supplier and used to update the Oracle Retail supplier address information.

Five different types of supplier addresses can be changed via this EDI interface. They are: business, postal, returned to, pick up and payment mailing address. This program always assumes that address information is primary for the address type.

### Scheduling Constraints

| Schedule Information      | Description                   |
|---------------------------|-------------------------------|
| Processing Cycle          | Phase 2 (Daily) or as needed. |
| Scheduling Considerations | N/A                           |
| Pre-Processing            | N/A                           |
| Post-Processing           | N/A                           |
| Threading Scheme          | N/A – File-based processing   |

### Restart/Recovery

The program uses non-fatal error handling to process input files. There is not enough volume to warrant the use of restart/recovery. A commit does not occur until the end of file processing and therefore if fatal errors are encountered updates are not committed and the program can restart without recovery.

### Locking Strategy

N/A

### Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table   | Select | Insert | Update | Delete |
|---------|--------|--------|--------|--------|
| SUPS    | Yes    | No     | No     | No     |
| STATE   | Yes    | No     | No     | No     |
| COUNTRY | Yes    | No     | No     | No     |
| ADDR    | Yes    | Yes    | Yes    | No     |

## I/O Specification

### Input File Layout

| Record Name | Field Name             | Field Type | Default Value | Description                                                                                             |
|-------------|------------------------|------------|---------------|---------------------------------------------------------------------------------------------------------|
| FHEAD       | File record descriptor | Char(5)    | FHEAD         | Describes file line type                                                                                |
|             | Line number            | Number(10) | 0000000001    | Sequential file line number                                                                             |
|             | Gentran_id             | Char(5)    | UPADD         | Identifies the file type                                                                                |
|             | File create date       | Char(14)   |               | YYYYMMDDHH24MISS format                                                                                 |
| FDETL       | File record descriptor | Char(5)    | FDETL         | Describes file line type                                                                                |
|             | Line number            | Number(10) |               | Sequential file line number                                                                             |
|             | Transaction number     | Number(10) |               | Sequential transaction number                                                                           |
|             | Add or Update          | Char(1)    |               | 'A'dd or 'U'pdate address                                                                               |
|             | Address type           | Char(2)    |               | Addr.addr_type:<br>01 – Business<br>02 – Postal<br>03 – returns<br>04 – Pick Up (Order)<br>05 – Payment |
|             | Supplier               | Number(10) |               | Sups.supplier                                                                                           |
|             | Address line 1         | Char(240)  |               | Addr.add_1                                                                                              |
|             | Address line 2         | Char(240)  |               | Addr.add_2                                                                                              |
|             | Address line 3         | Char(240)  |               | Addr.add_3                                                                                              |
|             | Contact name           | Char(120)  |               | Addr.contact_name                                                                                       |

| Record Name | Field Name             | Field Type | Default Value | Description                                                 |
|-------------|------------------------|------------|---------------|-------------------------------------------------------------|
|             | Contact Phone          | Char(20)   |               | Addr.contact_phone                                          |
|             | Contact fax            | Char(20)   |               | Addr.contact_fax                                            |
|             | City                   | Char(120)  |               | Addr.city                                                   |
|             | State                  | Char(3)    |               | Addr.state                                                  |
|             | Postal code            | Char(30)   |               | Addr.post                                                   |
|             | Country                | Char(3)    |               | Addr.country_id                                             |
| FTAIL       | File record descriptor | Char(5)    |               | Describes file record type                                  |
|             | Line number            | Number(10) |               | Sequential file line number (total number of lines in file) |
|             | Number of transactions | Number(10) |               | Number of transactions in file                              |

## ediupavl (Supplier Availability for Contracts Upload)

### Functional Area

EDI – Contracts

### Module Affected

EDIUPAVL.PC

### Design Overview

This module runs to upload a supplier availability schedule, which is a list of the items that a supplier has available. It writes data contained in this file to the RMS' SUP\_AVAIL table. This data is associated with the contracts functionality.

### Scheduling Constraints

| Schedule Information      | Description                 |
|---------------------------|-----------------------------|
| Processing Cycle          | Phase 1 (Daily)             |
| Scheduling Considerations | N/A                         |
| Pre-Processing            | N/A                         |
| Post-Processing           | N/A                         |
| Threading Scheme          | N/A – File-based processing |

### Restart/Recovery

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table     | Select | Insert | Update | Delete |
|-----------|--------|--------|--------|--------|
| SUP_AVAIL | No     | Yes    | Yes    | No     |

**I/O Specification****Input File Layout**

| Record Name | Field Name         | Field Type | Default Value | Description                                               |
|-------------|--------------------|------------|---------------|-----------------------------------------------------------|
| FHEAD       | Record descriptor  | Char(5)    | FHEAD         | Describes file line type                                  |
|             | Line number        | Number(10) | 0000000001    | Sequential file line number                               |
|             | File type          | Char(4)    | SPAV          |                                                           |
|             | Create date        | Char(14)   |               | File create date<br>YYYYMMDDHH24<br>MISS format           |
| FDETL       | Record descriptor  | Char(5)    | FDETL         | Describes file line type                                  |
|             | Line number        | Number(10) |               | Sequential file line number                               |
|             | Transaction number | Number(14) |               | Sequential transaction number                             |
|             | Supplier           | Number(10) |               |                                                           |
|             | Item type          | Char(3)    |               | 'ITM', 'UPC', or 'VPN'<br>item type                       |
|             | Item id            | Char(30)   |               | Actual Item/UPC/VPN<br>number                             |
|             | Item supplement    | Char(5)    |               | UPC supplement                                            |
|             | Available quantity | Number(12) |               | Available quantity*10000<br>(4 implied decimal<br>Places) |

| Record Name | Field Name               | Field Type | Default Value | Description                                         |
|-------------|--------------------------|------------|---------------|-----------------------------------------------------|
| FTAIL       | Record descriptor        | Char(5)    | FTAIL         | Describes file line type                            |
|             | Line number              | Number(10) |               | Sequential file line number (total # lines in file) |
|             | Number of detail records | Number(10) |               | Number of FDETL lines in file                       |

## ediupcat (New and Changed Upload from Supplier)

### Functional Area

EDI – Suppliers

### Module Affected

EDIUPCAT.PC

### Design Overview

The purpose of the EDIUPCAT batch program is to update the EDI\_NEW\_ITEM and EDI\_COST\_CHANGE tables. This allows the users to view and implement the vendor changes online instead of manually viewing and inserting information.

The input file format is in a Oracle Retail standard file format, rather than EDI format. The translation from EDI 888 and EDI 879 (unit cost and case cost) to this standard format is done by customers using an EDI translation product such as the Gentran translator.

For additional information about cost changes, see the chapter “Cost Change Batch” in this volume of the RMS Operations Guide.

### Scheduling Constraints

| Schedule Information      | Description                   |
|---------------------------|-------------------------------|
| Processing Cycle          | Phase 2 (Daily) or as needed. |
| Scheduling Considerations | N/A                           |
| Pre-Processing            | N/A                           |
| Post-Processing           | N/A                           |
| Threading Scheme          | N/A – File-based processing   |

### Restart/Recovery

File-based restart/recovery is used. Each file detail is committed to the database separately.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                          | Select | Insert | Update | Delete |
|--------------------------------|--------|--------|--------|--------|
| PERIOD                         | Yes    | No     | No     | No     |
| SUPS                           | Yes    | No     | No     | No     |
| ITEM_MASTER                    | Yes    | No     | No     | No     |
| ITEM_SUPPLIER                  | Yes    | No     | No     | No     |
| EDI_NEW_ITEM                   | Yes    | Yes    | Yes    | No     |
| PACKITEM                       | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY              | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC          | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_BRACKET_COST | Yes    | No     | No     | No     |
| V_DIFF_GROUP_HEAD              | Yes    | No     | No     | No     |
| DIFF_GROUP_DETAIL              | Yes    | No     | No     | No     |
| DIFF_IDS                       | Yes    | No     | No     | No     |
| ITEM_ZONE_PRICE                | Yes    | No     | No     | No     |
| SUP_BRACKET_COST               | Yes    | No     | No     | No     |
| WH                             | Yes    | No     | No     | No     |
| STORE                          | Yes    | No     | No     | No     |
| ITEM_LOC                       | Yes    | No     | No     | No     |
| EDI_COST_LOC                   | No     | Yes    | Yes    | No     |
| EDI_COST_CHG                   | No     | Yes    | Yes    | No     |

**I/O Specification****Input File Layout**

| Record Name | Field Name                  | Field Type | Default Value | Description                 |
|-------------|-----------------------------|------------|---------------|-----------------------------|
| FHEAD       | File Type Record Descriptor | Char(5)    | FHEAD         | Identifies file record type |



| Record Name | Field Name                  | Field Type  | Default Value      | Description                                                                                                                    |
|-------------|-----------------------------|-------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------|
|             | File Line Identifier        | Number(10)  | Sequential number. | Sequential file line number                                                                                                    |
|             | File Type Definition        | Char(4)     | UCAT               | Identifies file type.                                                                                                          |
|             | File Create Date            | Char(14)    | Create date        | Date on which the file was created in 'YYYYMMDDHH24MISS' format                                                                |
| FDETL       | File Type Record Descriptor | Char(5)     | FDETL              | Identifies file record type                                                                                                    |
|             | File Line Identifier        | Numeric(10) | Sequential number. | Sequential file line number                                                                                                    |
|             | Transaction sequence        | Number(10)  |                    | Sequential transaction number                                                                                                  |
|             | Supplier                    | Number(10)  |                    | Supplier number                                                                                                                |
|             | Supplier Name               | Char(240)   |                    | Supplier name                                                                                                                  |
|             | Duns Number                 | Number(9)   |                    | Dun and Bradstreet number identifies the supplier. Note the Duns Number and Duns Loc together, uniquely identifies a supplier. |
|             | Duns Loc                    | Number(4)   |                    | Dun and Bradstreet number identifies the location of the supplier.                                                             |
|             | Item                        | Char(25)    |                    | Item ID (blank if none)                                                                                                        |
|             | Ref item                    | Char(25)    |                    | Reference Item. For example, UPC (blank if none).                                                                              |
|             | Ref item type               | Char(6)     |                    | Reference item type.                                                                                                           |
|             | Item Parent                 | Char (25)   |                    | Item Parent which uniquely identifies the item/group at the level above the item.                                              |
|             | Parent VPN                  | Char(30)    |                    | Vendor product number of the parent item                                                                                       |
|             | VPN                         | Char(30)    |                    | Vendor product number of the item (blank if none)                                                                              |

| Record Name | Field Name                     | Field Type | Default Value | Description                                                                                                                                                                                                        |
|-------------|--------------------------------|------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Supplier item differentiator 1 | Char(120)  |               | Item differentiator description 1 (for example, color, size, descriptions). This field is displayed later when entering the item into RMS to use as a basis for choosing an appropriate differentiator within RMS. |
|             | Supplier item differentiator 2 | Char(120)  |               | Item differentiator description 2.                                                                                                                                                                                 |
|             | Supplier item differentiator 3 | Char(120)  |               | Item differentiator description 3.                                                                                                                                                                                 |
|             | Supplier item differentiator 4 | Char(120)  |               | Item differentiator description 4.                                                                                                                                                                                 |
|             | Item description               | Char(250)  |               | Item description                                                                                                                                                                                                   |
|             | Short description              | Char(120)  |               | Item short description for point of sales.                                                                                                                                                                         |
|             | Effective date                 | Char(14)   |               | Effective date in YYYYMMDDHH24MISS format                                                                                                                                                                          |
|             | Min order qty                  | Number(12) |               | Minimum order quantity * 10000 (4 implied decimal places)                                                                                                                                                          |
|             | Max order qty                  | Number(12) |               | Maximum order quantity * 10000 (4 implied decimal places)                                                                                                                                                          |
|             | Lead time                      | Number(4)  |               | Days from PO receipt to shipment                                                                                                                                                                                   |
|             | Unit cost                      | Number(20) |               | Unit cost* 10000 (4 implied decimal places)                                                                                                                                                                        |
|             | Gross unit weight              | Number(12) |               | Gross unit weight * 10000 (4 implied decimal places). The gross numeric value of weight per unit.                                                                                                                  |
|             | Net unit weight                | Number(12) |               | Net unit weight * 10000 (4 implied decimal places). The net numeric value of weight per unit.                                                                                                                      |
|             | Unit weight UOM                | Char(4)    |               | Item unit weight unit of measure                                                                                                                                                                                   |

| Record Name | Field Name             | Field Type | Default Value | Description                                                            |
|-------------|------------------------|------------|---------------|------------------------------------------------------------------------|
|             | Unit length            | Number(12) |               | Item unit length * 10000 (4 implied decimal places)                    |
|             | Unit width             | Number(12) |               | Item unit width * 10000 (4 implied decimal places)                     |
|             | Unit height            | Number(12) |               | Item unit height * 10000 (4 implied decimal places)                    |
|             | Unit lwh UOM           | Char(4)    |               | Item unit dimension unit of measure.                                   |
|             | Unit liquid volume     | Number(12) |               | Item unit liquid volume or capacity * 10000 (4 implied decimal places) |
|             | Unit liquid volume UOM | Char(4)    |               | Unit of measure of the item liquid volume/capacity                     |
|             | Case ref item          | Char(25)   |               | Case reference number. For example: case UPC code.                     |
|             | Case ref item type     | Char(6)    |               | Case reference number type. (blank if none).                           |
|             | Case item desc         | Char(250)  |               | Case item description                                                  |
|             | Case cost              | Number(20) |               | Case Cost * 10000 (4 implied decimal places)                           |
|             | Gross case weight      | Number(12) |               | Gross weight of the case* 10000 (4 implied decimal places)             |
|             | Net case weight        | Number(12) |               | Net weight of the case * 10000 (4 implied decimal places)              |
|             | Case weight UOM        | Char(4)    |               | Unit of measure of the case weight                                     |
|             | Case length            | Number(12) |               | Case length * 10000 (4 implied decimal places)                         |
|             | Case width             | Number(12) |               | Case width * 10000 (4 implied decimal places)                          |
|             | Case height            | Number(12) |               | Case height * 10000 (4 implied decimal places)                         |
|             | Case lwh UOM           | Char(4)    |               | Case dimension unit of measure.                                        |

| Record Name | Field Name             | Field Type | Default Value | Description                                                                                                                                                        |
|-------------|------------------------|------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Case liquid volume     | Number(12) |               | Case liquid volume or capacity * 10000 (4 implied decimal places)                                                                                                  |
|             | Case liquid volume UOM | Char(4)    |               | Unit of measure of the case liquid volume/capacity                                                                                                                 |
|             | Gross pallet weight    | Number(12) |               | Gross pallet weight* 10000 (4 implied decimal places)                                                                                                              |
|             | Net pallet weight      | Number(12) |               | Net pallet weight * 10000 (4 implied decimal places)                                                                                                               |
|             | Pallet weight UOM      | Char(4)    |               | Unit of measure of the pallet weight                                                                                                                               |
|             | Pallet length          | Number(12) |               | Pallet length* 10000 (4 implied decimal places)                                                                                                                    |
|             | Pallet width           | Number(12) |               | Pallet width* 10000 (4 implied decimal places)                                                                                                                     |
|             | Pallet height          | Number(12) |               | Pallet height * 10000 (4 implied decimal places)                                                                                                                   |
|             | Pallet lwh UOM         | Char(4)    |               | Pallet dimension unit of measure.                                                                                                                                  |
|             | Ti                     | Number(12) |               | Shipping units (cases) in one tier of a pallet * 10000 (4 implied decimal places)                                                                                  |
|             | Hi                     | Number(12) |               | Number of tiers in a pallet (height). * 10000 (4 implied decimal places)                                                                                           |
|             | Pack Size              | Number(12) |               | Supplied pack size * 10000 that is, Number of eaches per case pack. This is the quantity that orders must be placed in multiples of for the supplier for the item. |
|             | Inner pack size        | Number(12) |               | Supplied inner pack size* 10000 that is, Number of eaches per inner container.                                                                                     |
|             | Origin Country ID      | Char(3)    |               | Supplied origin country ID.                                                                                                                                        |
|             | Standard UOM           | Char(4)    |               | Unit of measure in which stock of the item is tracked at a corporate level.                                                                                        |

| Record Name | Field Name            | Field Type    | Default Value | Description                                                                                                                                                                                                                                                                                                                                                                        |
|-------------|-----------------------|---------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | UOM Conversion Factor | Number(20)    |               | Conversion Factor, 10 implied decimal places. Conversion factor between an "Each" and the standard_uom when the standard_uom is not in the quantity class (that is, if standard_uom = lb and 1 lb = 10 eaches, the factor is 10). This factor is used to convert sales and stock data when an item is retailed in eaches but does not have eaches as its standard unit of measure. |
|             | Packing Method        | Char(6)       |               | Packing Method code (HANG,FLAT)                                                                                                                                                                                                                                                                                                                                                    |
|             | Location              | Number(10)    |               | Location that the supplier distributes to or this may be a number used by the supplier to identify a non-Oracle Retail location.                                                                                                                                                                                                                                                   |
|             | Location Type         | Char(1)       |               | This field contains the type of location ('S' for store and 'W' for warehouse).                                                                                                                                                                                                                                                                                                    |
|             | Bracket Value 1       | Number (12,4) |               | This contains the primary bracket value of the supplier.                                                                                                                                                                                                                                                                                                                           |
|             | Bracket UOM 1         | Char(4)       |               | This field contains the unit of measure of the primary bracket.                                                                                                                                                                                                                                                                                                                    |
|             | Bracket Type 1        | Char (6)      |               | This field contains the UOM class.                                                                                                                                                                                                                                                                                                                                                 |
|             | Bracket Value 2       | Number (12,4) |               | This field contains the secondary bracket value for the supplier.                                                                                                                                                                                                                                                                                                                  |
|             | Unit cost new         | Number (20,4) |               | This field contains the new unit cost of the bracket.                                                                                                                                                                                                                                                                                                                              |
|             | Case Bracket Value 1  | Number (12,4) |               | This field contains the primary bracket value of the supplier for a case UPC.                                                                                                                                                                                                                                                                                                      |

| Record Name | Field Name                  | Field Type    | Default Value | Description                                                                                                                                                                                                                                 |
|-------------|-----------------------------|---------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Case Bracket UOM 1          | Char(4)       |               | This field contains the unit of measure of the primary bracket for a case UPC.                                                                                                                                                              |
|             | Case Bracket Type 1         | Char (6)      |               | This field contains the UOM class for a case UPC.                                                                                                                                                                                           |
|             | Case Bracket Value 2        | Number (12,4) |               | This field contains the secondary bracket value for the supplier for a case UPC.                                                                                                                                                            |
|             | Case Unit cost new          | Number (20,4) |               | This field contains the new unit cost of the bracket for a case UPC.                                                                                                                                                                        |
|             | Item_diff_1                 | Char(10)      |               | This field holds a unique number (identifier) of the differential types.<br>(For example, diff_type = 'S' might have these IDs: 1, 50, 1000; then diff_type= 'C' cannot use the same numbers; the IDs have to be different: 2, 20,51, 1001) |
|             | Item_diff_2                 | Char(10)      |               | As above                                                                                                                                                                                                                                    |
|             | Item_diff_3                 | Char(10)      |               | As above                                                                                                                                                                                                                                    |
|             | Item_diff_4                 | Char(10)      |               | As above                                                                                                                                                                                                                                    |
|             | unit_retail                 | Number(20,4)  |               | This field contains the suppliers recommended retail value for the item.                                                                                                                                                                    |
|             | retail_zone_group           | Number(4)     |               | This field contains the retail zone group number of the given item.                                                                                                                                                                         |
|             | consignment_rate            | Number(12,4)  |               | This field contains the consignment rate for this item for the supplier.                                                                                                                                                                    |
|             | break_to_sell_ind           | Char(1)       |               | Indicates whether item is a break to sell item.                                                                                                                                                                                             |
| FTAIL       | Item_level                  | Char(1)       |               | Indicates the items' item_level                                                                                                                                                                                                             |
|             | Tran_level                  | Char(1)       |               | Indicates the items' tran_level                                                                                                                                                                                                             |
|             | File Type Record Descriptor | Char(5)       | FTAIL         | Identifies file record type                                                                                                                                                                                                                 |

| Record Name | Field Name           | Field Type     | Default Value       | Description                                                  |
|-------------|----------------------|----------------|---------------------|--------------------------------------------------------------|
|             | File Line Identifier | Numeric(10)    | Sequential number.. | Sequential file line number                                  |
|             | File Record Counter  | Numeric ID(10) |                     | Number of records/transactions processed in the current file |





## Forecasting Batch

### Overview

The batch programs in this chapter facilitate processing related to forecasting data and to forecasting systems.

The programs that begin with the letters FCS enable the roll up of item demand forecasting data to the subclass, class, or department level. The programs also provide a mechanism to purge data from the forecasting tables.

---

**Note:** RMS 12.0 is **not** integrated with Oracle Retail planning and forecasting systems (for example, Oracle Retail Demand Forecasting).

---

### Batch Design Summary

The following batch designs are included in this functional area:

- FCSTPRG.PC (Oracle Retail Demand Forecasting Purge)
- FCSTRBLD.PC (Oracle Retail Demand Forecasting Rollup)
- FCSTRBLD\_SBC.PC (Oracle Retail Demand Forecasting Rollup by Department, Class and Subclass)
- FTMEDNLD.PC (Time Hierarchy Download)
- SOUTDNLD.PC (Stockout Download)

### fcstprg (Oracle Retail Demand Forecasting Purge)

#### Functional Area

Demand Forecasting

#### Module Affected

FCSTPRG.PC

#### Design Overview

This program deletes data from forecast information tables. Data deletion is performed by partition truncation, table truncation or deletion by domain. The method of deletion is dependent on whether or not the table is partitioned. This program serves to delete data by domains so that they can re-loaded with new forecast information from a forecasting system.

This program must be run as either the RMS schema owner, or be run by a user that has been granted the following system privileges:

‘drop any table’

‘alter any table’

#### Scheduling Constraints

| Schedule Information      | Description                             |
|---------------------------|-----------------------------------------|
| Processing Cycle          | AD-HOC                                  |
| Scheduling Considerations | N/A                                     |
| Pre-Processing            | prepost fcstprg pre - disables indexes  |
| Post-Processing           | prepost fcstprg post - rebuilds indexes |
| Threading Scheme          | N/A                                     |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| ITEM_FORECAST           | No     | No     | No     | Yes    |
| DEPT_SALES_FORECAST     | No     | No     | No     | Yes    |
| CLASS_SALES_FORECAST    | No     | No     | No     | Yes    |
| SUBCLASS_SALES_FORECAST | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**fcstrbld (Oracle Retail Demand Forecasting Rollup)****Functional Area**

Demand Forecasting

**Module Affected**

FCSTRBLD.PC

## Design Overview

This program is designed to roll-up new or updated forecasted unit sales data from the item\_forecast table. This data will be summarized into the subclass, class and department level sales forecast tables.

## Scheduling Constraints

| Schedule Information      | Description                                                  |
|---------------------------|--------------------------------------------------------------|
| Processing Cycle          | Phase 3 (weekly)                                             |
| Scheduling Considerations | N/A                                                          |
| Pre-Processing            | N/A                                                          |
| Post-Processing           | prepost fcstrbld post – truncates the FORECAST_REBUILD table |
| Threading Scheme          | Threaded by domain id                                        |

## Restart/Recovery

The logical unit of work is a domain id. The program commits each time the rollups (dept, class and subclass) for a domain id is successfully processed.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| FORECAST_REBUILD        | Yes    | No     | No     | Yes    |
| SUBCLASS_SALES_FORECAST | Yes    | Yes    | No     | No     |
| ITEM_MASTER             | Yes    | No     | No     | No     |
| ITEM_FORECAST           | Yes    | No     | No     | No     |
| STORE                   | Yes    | No     | No     | No     |
| CLASS_SALES_FORECAST    | Yes    | Yes    | No     | No     |
| DEPT_SALES_FORECAST     | Yes    | Yes    | No     | No     |

## I/O Specification

N/A

## fcstrbld\_sbc (Oracle Retail Demand Forecasting Rollup by Department, Class and Subclass)

### Functional Area

Demand Forecasting

### Module Affected

FCSTRBLD\_SBC.PC

### Design Overview

The module rolls up the sales forecast data at subclass and class level to class and department level respectively and inserts the data. The program selects records from the table SUBCLASS\_SALES\_FORECAST and writes the records to CLASS\_SALES\_FORECAST and selects the data from CLASS\_SALES\_FORECAST and writes into DEPT\_SALES\_FORECAST using the domain ID stored in the table FORECAST\_REBUILD. The record in FORECAST\_REBUILD is deleted after the record is written to the above destination tables.

### Scheduling Constraints

| Schedule Information      | Description                      |
|---------------------------|----------------------------------|
| Processing Cycle          | Phase 3 (Weekly)                 |
| Scheduling Considerations | After completion of FCSTRBLD.PC. |
| Pre-Processing            | N/A                              |
| Post-Processing           | N/A                              |
| Threading Scheme          | N/A                              |

### Restart/Recovery

Restart/recovery is based on the values stored in restart\_bookmark from the last commit prior to failure. The values are for the last domain\_id that was not rolled up completely.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| FORECAST_REBUILD     | Yes    | No     | No     | Yes    |
| CLASS_SALES_FORECAST | Yes    | Yes    | No     | No     |

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| DEPT_SALES_FORECAST     | No     | Yes    | No     | No     |
| SUBCLASS_SALES_FORECAST | Yes    | No     | No     | No     |
| STORE                   | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**ftmednld (Time Hierarchy Download)****Functional Area**

RMS-planning system interface

**Module Affected**

FTMEDNLD.PC

**Design Overview**

The FTMEDNLD.PC module downloads the RMS calendar (year, half, quarter, month, week, day, and date) in the 454-calendar format. The download consists of the entire calendar in the RMS. This program accounts for a fiscal year that could be different from the standard year in the CALENDAR table.

The extracted flat file needs to be transferred to a location where the planning system (with its transformation script) can access it.

**Scheduling Constraints**

| Schedule Information      | Description         |
|---------------------------|---------------------|
| Processing Cycle          | Ad hoc              |
| Scheduling Considerations | N/A                 |
| Pre-Processing            | N/A                 |
| Post-Processing           | N/A                 |
| Threading Scheme          | N/A (Single Thread) |

**Restart/Recovery**

Due to the relatively small amount of processing this program performs; restart recovery will not be used. The calls to retek\_init() and retek\_close() are used in the program only for logging purposes (to prevent double-runs).

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| CALENDAR       | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |

**I/O Specification****Output Files**

The file outputted will be named rmse\_clndmstr.dat.

| Record Name | Field Name | Field Type | Default Value | Description                                                        |
|-------------|------------|------------|---------------|--------------------------------------------------------------------|
|             | Year       | Number(4)  |               | The 4-5-4 year                                                     |
|             | Half       | Number(1)  |               | The 4-5-4 half of the year, valid values are 1 or 2                |
|             | Quarter    | Number(1)  |               | The 4-5-4 quarter of the year, valid values 1-4                    |
|             | Month      | Number(2)  |               | The 4-5-4 month of the year, valid values 1-12                     |
|             | Week       | Number(2)  |               | The 4-5-4 week of the year, valid values 1-53                      |
|             | Day        | Number(1)  |               | The 4-5-4 day of the current week, valid values 1-7                |
|             | Date       | Date       |               | The date from which the 4-5-4 data was derived, in YYYYMMDD format |

**soutdnld (Stockout Download)****Functional Area**

RMS to a forecasting interface

**Module Affected**

SOUTDNLD.PC

**Design Overview**

A forecasting interface requires a notification whenever an item's or store's stock on hand goes to zero or below that level. This SOUTDNLD program loops through the item/store table and outputs an item/store combination that has a stock out condition to an output file. This output file will then be sent to the forecasting system.

## Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                               |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 4 (Daily)                                                                                                                                                                           |
| Scheduling Considerations | Any processing that updates the stock levels should be completed before running this program.                                                                                             |
| Pre-Processing            | N/A                                                                                                                                                                                       |
| Post-Processing           | N/A                                                                                                                                                                                       |
| Threading Scheme          | The forecasting system requires that the output files generated by this program be grouped by domain number. To accommodate this requirement, SOUTDNLD.PC should be threaded by a domain. |

## Restart/Recovery

The logical unit of work for this program is set at item/location level. Table based restart/recovery is used. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O.

Since threads are determined by the value of the domain ID, the RESTART\_PROGRAM\_STATUS table should contain a row for each domain ID. The thread value of the domain ID should be used as the thread value on this table. The total number of domains/number of threads should be equal to the number of rows on the RESTART\_PROGRAM\_STATUS table. This value must be entered into the restart\_control table num\_threads field. Note that anytime a new domain is created, an additional row should be added to the RESTART\_PROGRAM\_STATUS table with the thread value equal to the domain ID and the restart\_control table num\_threads field must be incremented to equal the total number of domains.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| ITEM_MASTER     | Yes    | No     | No     | No     |
| ITEM_LOC_SOH    | Yes    | No     | No     | No     |
| DOMAIN_DEPT     | Yes    | No     | No     | No     |
| DOMAIN_CLASS    | Yes    | No     | No     | No     |
| DOMAIN_SUBCLASS | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Field Name | Field Type | Default Value | Description                                                                                |
|------------|------------|---------------|--------------------------------------------------------------------------------------------|
| Date       | Char(8)    | Period.vdate  | The date of the stockout in YYYYMMDD format.                                               |
| Store      | Number(10) |               | The store at which the sku encountered the stockout – left justified with trailing blanks. |
| Item       | Char(25)   |               | The item that encountered the stockout – left justified with trailing blanks.              |



## General Ledger (GL) Batch

### Overview

RMS stages GL data for subsequent upload into a financial system. A set of batch processes gather and organize the data before using it to populate the staging table, STG\_FIF\_GL\_DATA.

### Batch Design Summary

The following batch designs are included in this functional area:

- DEALFINC.PC (Deal Fixed Income)
- FIFGLDN1.PC (Financial General Ledger Download 1)
- FIFGLDN2.PC (Financial General Ledger Download 2)
- FIFGLDN3.PC (Financial General Ledger Download 3)

### dealfinc (Deal Fixed Income)

#### Functional Area

Deals maintenance

#### Module Affected

DEALFINC.PC

#### Design Overview

This module writes to the STG\_FIF\_GL\_DATA financial staging table to perform stock ledger processing for fixed deals. It splits deal income over all dept/class/subclass locations on the deal. This prorated income is written to the general ledger under a suitable cost center mapping. Each merchandise level/location on the deal has an associated contrib\_ratio to determine how much of the total amount is apportioned to items in that merchant/location. Because the user could have entered any fraction of 1 into these fields, the contrib\_ratios probably does not add up to 1. Therefore, the ratios are prorated across all locations so they add up to 1. This value is then apportioned between all subclasses for the general ledger.

#### Scheduling Constraints

| Schedule Information      | Description                                                                  |
|---------------------------|------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3, Daily                                                               |
| Scheduling Considerations | Should be run after DEALACT.PC, before DEALFCT.PC, DEALDAY.PC and SALMTH.PC. |
| Pre-Processing            | N/A                                                                          |

| Schedule Information | Description              |
|----------------------|--------------------------|
| Post-Processing      | N/A                      |
| Threading Scheme     | Multithreaded on Deal ID |

**Restart/Recovery**

The logical unit of work for this program is a deal\_id. The database commit takes place when number of deal records processed is equal to the commit max counter in the restart control table.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| FIF_GL_SETUP         | Yes    | No     | No     | No     |
| FIXED_DEAL           | Yes    | No     | No     | No     |
| FIXED_DEAL_DATES     | Yes    | No     | No     | No     |
| FIXED_DEAL_MERCH     | Yes    | No     | No     | No     |
| FIXED_DEAL_MERCH_LOC | Yes    | No     | No     | No     |
| SUBCLASS             | Yes    | No     | No     | No     |
| STORE                | Yes    | No     | No     | No     |
| WH                   | Yes    | No     | No     | No     |
| FIF_GL_CROSS_REF     | Yes    | No     | No     | No     |
| STG_FIF_GL_DATA      | No     | Yes    | No     | No     |

**I/O Specification**

N/A

**fifglDn1 (General Ledger Interface)****Functional Area**

General Ledger (GL) batch

**Module Affected**

FIFGLDN1.PC

## Design Overview

This program extracts the detailed stock ledger information for certain transaction types on a daily basis in order to bridge the information to an interfaced financial application.

The program reads from the IF\_TRAN\_DATA table for each transaction type/amount type and posts it to the Oracle Retail general ledger table (FIF\_GL\_DATA) at the SKU detail level along with the reference fields. These reference fields are bridged to the financial application for segregation of different transactions (for example, stock adjustment codes, mapping transfers to transfer numbers, and so on).

## Scheduling Constraints

| Schedule Information      | Description                                     |
|---------------------------|-------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                 |
| Scheduling Considerations | Should run after SALSTAGE and prior to SALAPND. |
| Pre-Processing            | N/A                                             |
| Post-Processing           | prepost FIFGLDN1 post                           |
| Threading Scheme          | Threaded by department                          |

## Restart/Recovery

The logical unit of work is department/class/subclass. The batch is multithreaded using the v\_restart\_dept view.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| STORE            | Yes    | No     | No     | No     |
| WH               | Yes    | No     | No     | No     |
| PARTNER          | Yes    | No     | No     | No     |
| IF_TRAN_DATA     | Yes    | No     | No     | No     |
| CODE_DETAIL      | Yes    | No     | No     | No     |
| FIF_GL_CROSS_REF | Yes    | No     | No     | No     |
| FIF_GL_SETUP     | Yes    | No     | No     | No     |
| STG_FIF_GL_DATA  | No     | Yes    | No     | No     |

**I/O Specification**

NA

**fifgldn2 (General Ledger Interface)****Functional Area**

General Ledger (GL) batch

**Module Affected**

FIFGLDN2.PC

**Design Overview**

This program summarizes stock ledger data from the transaction staging table (IF\_TRAN\_DATA) based on the level of information required and writes it to the financial general ledger staging table. The transactions extracted are determined by the code\_type 'GLRT' (general ledger rolled transactions). Written information is then sent to the financial applications for general ledger purposes. Stock ledger information may be rolled-up at department, class or subclass level. The level at which information is rolled-up to is determined by the gl\_rollup field on the SYSTEM\_OPTIONS table.

**Scheduling Constraints**

| Schedule Information      | Description                                     |
|---------------------------|-------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                 |
| Scheduling Considerations | Should run after SALSTAGE and prior to SALAPND. |
| Pre-Processing            | N/A                                             |
| Post-Processing           | N/A                                             |
| Threading Scheme          | Threaded by department                          |

**Restart/Recovery**

The logical unit of work is dependent on the level of rollup defined in system\_options.gl\_rollup. It can be department (department rollup), department/class (class rollup) or department/class/subclass (subclass rollup). The batch is multithreaded using the v\_restart\_dept view.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| STORE            | Yes    | No     | No     | No     |
| WH               | Yes    | No     | No     | No     |
| PARTNER          | Yes    | No     | No     | No     |
| IF_TRAN_DATA     | Yes    | No     | No     | No     |
| CODE_DETAIL      | Yes    | No     | No     | No     |
| FIF_GL_CROSS_REF | Yes    | No     | No     | No     |
| FIF_GL_SETUP     | Yes    | No     | No     | No     |
| STG_FIF_GL_DATA  | No     | Yes    | No     | No     |

**I/O Specification**

NA

**fifgldn3 (General Ledger Interface)****Functional Area**

General Ledger (GL) batch

**Module Affected**

FIFGLDN3.PC

**Design Overview**

This program summarizes stock ledger data from the monthly stock ledger table (MONTH\_DATA) based on the level of information required and writes it to the financial general ledger staging table. The transactions extracted are determined by the code\_type 'GLRT' (general ledger rolled transactions). Written information is then sent to the financial applications for general ledger purposes. Stock ledger information may be rolled-up at department, class or subclass level. The level at which information is rolled-up to is determined by the gl\_rollup field on the SYSTEM\_OPTIONS table.

**Scheduling Constraints**

| Schedule Information      | Description              |
|---------------------------|--------------------------|
| Processing Cycle          | Phase 3 (monthly)        |
| Scheduling Considerations | Should run after SALMTH. |
| Pre-Processing            | N/A                      |
| Post-Processing           | N/A                      |
| Threading Scheme          | Threaded by location     |

**Restart/Recovery**

The logical unit of work is dependent on the level of rollup defined in system\_options.gl\_rollup. It can be department (department rollup), department/class (class rollup) or department/class/subclass (subclass rollup). The batch is multithreaded using the v\_restart\_all\_locations view.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| STORE             | Yes    | No     | No     | No     |
| WH                | Yes    | No     | No     | No     |
| PARTNER           | Yes    | No     | No     | No     |
| MONTH_DATA        | Yes    | No     | No     | No     |
| CODE_DETAIL       | Yes    | No     | No     | No     |
| FIF_GL_CROSS_REF  | Yes    | No     | No     | No     |
| FIF_GL_SETUP      | Yes    | No     | No     | No     |
| TRAN_DATA_HISTORY | Yes    | No     | No     | No     |
| STG_FIF_GL_DATA   | No     | Yes    | No     | No     |

**I/O Specification**

NA

---

## Geocode Hierarchy Batch

### Overview

A geocode is the code that identifies a combination of the country, state, county, and city in which locations operate.

### Batch Design Summary

The following batch design is included in this functional area:

- GCUPLD.PC (Geocode Hierarchy Upload)

### gcupld (Geocode Hierarchy Upload)

#### Functional Area

Geocode hierarchy

#### Module Affected

GCUPLD.PC

#### Design Overview

A geocode identifies a combination of the country, state, county and city in which locations operate.

GCUPLD.PC (geocode hierarchy upload) provides the ability to upload geocodes from an outside source into RMS. This batch module lets retailers delete current geocodes and create new geocodes in the system. A flat file is used to feed the program the additions and deletions to the geocode tables. Validation determines if duplicate records exist, dependencies exist, and the flat file is in the correct format. If errors occur in the validation of the record, it is written out to a reject file to allow further investigation of the record.

#### Scheduling Constraints

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad Hoc      |
| Scheduling Considerations | Ad Hoc      |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

## Restart/Recovery

This is a file based upload and a file based restart/recovery logic. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O. The recommended commit counter setting is 10000 records (subject to change based on implementation).

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| GEOCODE_TEMP      | YES    | YES    | NO     | YES    |
| DISTRICT_GEOCODES | YES    | YES    | NO     | YES    |
| CITY_GEOCODES     | YES    | YES    | NO     | YES    |
| COUNTY_GEOCODES   | YES    | YES    | NO     | YES    |
| STATE_GEOCODES    | YES    | YES    | NO     | YES    |
| COUNTRY_GEOCODES  | YES    | YES    | NO     | YES    |
| GEOCODE_STORE     | YES    | NO     | NO     | NO     |
| GEOCODE_TXCDE     | YES    | NO     | NO     | NO     |

## I/O Specification

### Output File Layout

| Record Name | Field Name           | Field Type | Default Value | Description                               |
|-------------|----------------------|------------|---------------|-------------------------------------------|
| FHEAD       | File head descriptor | Char(5)    | FHEAD         | Describes the file line type              |
|             | Line id              | Char(10)   | 0000000001    | Sequential file line number               |
|             | Gentran ID           | Char(4)    | 'GCUP'        | Identifies which translation Gentran uses |
|             | Current date         | Char(14)   |               | File date in YYYYMMDDHH24MISS format      |



| Record Name | Field Name             | Field Type | Default Value | Description                                                                    |
|-------------|------------------------|------------|---------------|--------------------------------------------------------------------------------|
| FDETL       | File record descriptor | Char(5)    | FDETL         | Describes file line type                                                       |
|             | Line id                | Char(10)   |               | Sequential file line number                                                    |
|             | Country Geocode        | Char(4)    |               | Country Geocode                                                                |
|             | State Geocode          | Char(4)    |               | State Geocode                                                                  |
|             | County Geocode         | Char(4)    |               | County Geocode                                                                 |
|             | City Geocode           | Char(4)    |               | City Geocode                                                                   |
|             | District Geocode       | Char(4)    |               | District Geocode                                                               |
|             | Geocode Level          | Char(6)    |               | Geocode Level Valid values are:<br>'CNTRY','STATE','COUNTY',<br>'CITY', 'DIST' |
|             | Geocode Description    | Char(250)  |               | Geocode Description                                                            |
|             | Add Delete Ind         | Char(1)    |               | Add/delete Indicator<br>Valid values are: 'A', 'D'                             |
| FTAIL       | File record descriptor | Char(5)    | FTAIL         | Marks end of file                                                              |
|             | Line id                | Char(10)   |               | Sequential file line number                                                    |
|             | Number of lines        | Number(10) |               | Number of lines in file not counting FHEAD and FTAIL                           |



---

## Inventory Adjustment Batch

### Overview

The batch programs in this area address the deletion of obsolete inventory adjustment records where a pre-determined number of months have elapsed.

For more information about Inventory Adjustments, see the RMS 12.0 User Guide.

### Batch Design Summary

The following batch design is included in this functional area:

- INVAPRG.PC (Inventory Adjustment Purge)

### invaprg (Inventory Adjustment Purge)

#### Functional Area

Inventory Adjustment

#### Module Affected

INVAPRG.PC

#### Design Overview

The Inventory Adjustment Purge module deletes all obsolete inventory adjustment records whose adjustment date has elapsed a pre-determined number of months. The number of months that inventory adjustment records are kept before they are purged by this batch is defined in the SYSTEM\_OPTIONS table.

#### Scheduling Constraints

| Schedule Information      | Description      |
|---------------------------|------------------|
| Processing Cycle          | AD-HOC (monthly) |
| Scheduling Considerations | N/A              |
| Pre-Processing            | N/A              |
| Post-Processing           | N/A              |
| Threading Scheme          | N/A              |

#### Restart/Recovery

N/A

#### Locking Strategy

N/A

#### Performance Considerations

N/A

### Key Tables Affected

| Table        | Select | Insert | Update | Delete |
|--------------|--------|--------|--------|--------|
| UNIT_OPTIONS | Yes    | No     | No     | No     |
| PERIOD       | Yes    | No     | No     | No     |
| INV_ADJ      | No     | No     | No     | Yes    |

### Shared Modules

N/A

### I/O Specification

N/A

## Invoice Matching Batch

### Overview

RMS stages invoice records to be integrated into the Oracle Retail Invoice Matching (ReIM) product. It stages invoice records for: return to vendor (RTV), consignment, deals, trade management, obligations, and customs entry. This interface between these products does not have any functionality that is inherent to RMS.

### Batch Design Summary

The following batch designs are included in this functional area:

**Note:** The batch program, SAEXPIM.PC, has a functional connection to this chapter. For the design, please see the chapter, “Oracle Retail Sales Audit Batch,” in this volume of the Operations Guide.

- EDIDLINV.PC (EDI Download Invoice)
- INVCLSHP.PC (Invoice Close Shipments)
- INVPRG.PC (Invoice Purge)

### edidlinv (EDI Invoice Download)

#### Functional Area

EDI – Invoice Matching Interface

#### Module Affected

EDIDLINV.PC

#### Design Overview

The EDIDLINV program extracts invoice information from RMS invoice tables (INVC\_HEAD, INVC\_DETAIL) to a flat file. This flat file is read by EDI and uploaded to ReIM tables such as IM\_DOC\_HEAD, IM\_INVOICE\_DETAIL and IM\_DOC\_NON\_MERCH. This batch program is run daily, extracting invoice records whose invoice date falls on the current vdate.

#### Scheduling Constraints

| Schedule Information      | Description                |
|---------------------------|----------------------------|
| Processing Cycle          | Phase 4 (Daily)            |
| Scheduling Considerations | N/A                        |
| Pre-Processing            | N/A                        |
| Post-Processing           | N/A                        |
| Threading Scheme          | Multi-threaded by location |

**Restart/Recovery**

Restart/recovery for this program is set up at the invoice ID and line sequence level. The program resumes writing to file starting on the next line where the previous process ended.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| INVC_HEAD      | Yes    | No     | Yes    | No     |
| INVC_DETAIL    | Yes    | No     | No     | No     |
| INVC_XREF      | Yes    | No     | No     | No     |
| INVC_MERCH_VAT | Yes    | No     | No     | No     |
| INVC_NON_MERCH | Yes    | No     | No     | No     |
| ITEM_MASTER    | Yes    | No     | No     | No     |
| SUPS           | Yes    | No     | No     | No     |
| PARTNER        | Yes    | No     | No     | No     |
| VAT_CODE_RATES | Yes    | No     | No     | No     |
| PERIOD         | Yes    | No     | No     | No     |
| WH             | Yes    | No     | No     | No     |
| STORE          | Yes    | No     | No     | No     |

**I/O Specification****Output File Layout**

| Record Name | Field Name        | Field Type | Default Value | Description                                                         |
|-------------|-------------------|------------|---------------|---------------------------------------------------------------------|
| FHEAD       | Record descriptor | Char(5)    | FHEAD         | Describes file record type. Valid value is FHEAD.                   |
|             | Line id           | Number(10) | 0000000001    | Sequential file line number.                                        |
|             | Gentran ID        | Char(5)    | UPINV         | The type of transaction this file represents. Valid value is UPINV. |

| Record Name | Field Name                      | Field Type  | Default Value | Description                                                                                                                                                                                                                                                                                                                                                                                                        |
|-------------|---------------------------------|-------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Current date                    | Char(14)    |               | Vdate in YYYYMMDDHH24MISS format.                                                                                                                                                                                                                                                                                                                                                                                  |
| THEAD       | Record descriptor               | Char(5)     |               | Describes file record type. Valid value is THEAD.                                                                                                                                                                                                                                                                                                                                                                  |
|             | Line id                         | Number (10) |               | Sequential file line number.                                                                                                                                                                                                                                                                                                                                                                                       |
|             | Transaction number              | Number(10)  |               | Sequential transaction number. All records within this transaction also have this transaction number.                                                                                                                                                                                                                                                                                                              |
|             | Document Type                   | Char(6)     |               | Describes the type of document being uploaded. The document type determines the types of detail information that are valid for the document upload. Invoice types are held on the codes table under a code type of 'IMIT'.                                                                                                                                                                                         |
|             | Vendor Document Number          | Char (30)   |               | Vendor's document number.                                                                                                                                                                                                                                                                                                                                                                                          |
|             | Vendor Type                     | Char(6)     |               | Type of vendor (either supplier or partner) for this document. Valid values include Bank 'BK', Agent 'AG', Freight Forwarder 'FF', Importer 'IM', Broker 'BR', Factory 'FA', Applicant 'AP', Consolidator 'CO', Consignee 'CN', Supplier Hierarchy Level 1 'S1', Supplier Hierarchy Level 2 'S2', and Supplier Hierarchy Level 3 'S3'. These partner types are held on the codes table under the code_type 'PTAL'. |
|             | Vendor ID                       | Char(10)    |               | Vendor for this document.                                                                                                                                                                                                                                                                                                                                                                                          |
|             | Vendor Document Date            | Char(14)    |               | Date document was issued by the vendor (in YYYYMMDDHH24MISS format).                                                                                                                                                                                                                                                                                                                                               |
|             | Order Number / RTV order number | Number(10)  |               | Merchandising system order number for this document. Required for merchandise invoices and optional for others. This field can also contain the RTV order number if the RTV flag is 'Y'                                                                                                                                                                                                                            |

| Record Name | Field Name       | Field Type   | Default Value | Description                                                                                                                                                |
|-------------|------------------|--------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Location         | Number(10)   |               | Merchandising system location for this document.                                                                                                           |
|             | Location Type    | Char(1)      |               | Merchandising system location type (either 'S'tore or 'W'arehouse) for this document. Required for merchandise invoices and optional for others.           |
|             | Terms            | Char(15)     |               | Terms of this document. If terms are not provided, the vendor's default terms are associated with this record.                                             |
|             | Due Date         | Char(14)     |               | Date the amount due is due to the vendor (YYYYMMDDHH24MISS format). If due date is not provided, default due date is calculated based on vendor and terms. |
|             | Payment method   | Char(6)      |               | Method for paying this document.                                                                                                                           |
|             | Currency code    | Char(3)      |               | Currency code for all monetary amounts on this document.                                                                                                   |
|             | Exchange rate    | Number(12,4) |               | Exchange rate *10000 (implied 4 decimal places) for conversion of document currency to the primary currency.                                               |
|             | Sign Indicator   | Char(1)      |               | Indicates either a positive (+) or a negative (-) total cost amount.                                                                                       |
|             | Total Cost       | Number(20,4) |               | Total document cost *10000 (implied 4 decimal places), including all items and costs on this document. This value is in the document currency.             |
|             | Sign Indicator   | Char(1)      |               | Indicates either a positive (+) or a negative (-) total vat amount.                                                                                        |
|             | Total VAT Amount | Number(20,4) |               | Total VAT amount *10000 (implied 4 decimal places), including all items and costs on this document. This value is in the document currency.                |
|             | Sign Indicator   | Char(1)      |               | Indicates either a positive (+) or a negative (-) total quantity amount.                                                                                   |



| Record Name | Field Name                  | Field Type   | Default Value | Description                                                                                                                                            |
|-------------|-----------------------------|--------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Total Quantity              | Number(12,4) |               | Total quantity of items *10000 (implied 4 decimal places) on this document. This value is in EACHES (no other units of measure are supported in ReIM). |
|             | Sign Indicator              | Char(1)      |               | Indicates either a positive (+) or a negative (-) total discount amount.                                                                               |
|             | Total Discount              | Number(12,4) |               | Total discount *10000 (implied 4 decimal places) applied to this document. This value is in the document currency.                                     |
|             | Freight Type                | Char(6)      | NULL          | The freight method for this document. Always blank.                                                                                                    |
|             | Paid Ind                    | Char(1)      |               | Indicates if this document is paid.                                                                                                                    |
|             | Multi-Location              | Char(1)      | N             | Indicates if this invoice goes to multiple locations.                                                                                                  |
|             | Merchandise Type            | Char(1)      |               | Indicates if this invoice is a consignment invoice.                                                                                                    |
|             | Deal Id                     | Number(10)   | NULL          | Deal Id from RMS if this invoice is a deal bill back invoice. Always blank.                                                                            |
|             | Deal Approval Indicator     | Char(1)      | NULL          | Indicates if the document on IM_DOC_HEAD is to be created in Approved or Submitted status. Always blank.                                               |
|             | RTV indicator               | Char(1)      |               | Indicates if this invoice is a RTV invoice.                                                                                                            |
|             | Custom Document Reference 1 | Char(30)     | NULL          | This optional field is included in the upload file for client customization. No validation is performed on this field. Always blank.                   |
|             | Custom Document Reference 2 | Char(30)     | NULL          | This optional field is included in the upload file for client customization. No validation is performed on this field. Always blank.                   |

| Record Name | Field Name                      | Field Type   | Default Value | Description                                                                                                                          |
|-------------|---------------------------------|--------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------|
|             | Custom Document Reference 3     | Char(30)     | NULL          | This optional field is included in the upload file for client customization. No validation is performed on this field. Always blank. |
|             | Custom Document Reference 4     | Char(30)     | NULL          | This optional field is included in the upload file for client customization. No validation is performed on this field. Always blank. |
|             | Cross-reference document number | Number(10)   |               | Document that a credit note is for. Blank for all document types other than merchandise invoices.                                    |
| TDETL       | Record descriptor               | Char(5)      |               | Describes file record type. Valid value is TDETL.                                                                                    |
|             | Line id                         | Number(10)   |               | Sequential file line number.                                                                                                         |
|             | Transaction number              | Number(10)   |               | Transaction number for this item detail record.                                                                                      |
|             | UPC                             | Char(25)     | NULL          | UPC for this detail record. Valid item number is retrieved for the UPC. Always blank.                                                |
|             | UPC Supplement                  | Number(5)    | NULL          | Supplement for the UPC. Always blank.                                                                                                |
|             | Item                            | Char(25)     |               | Item for this detail record.                                                                                                         |
|             | Sign Indicator                  | Char(1)      |               | Indicates either a positive (+) or a negative (-) Original Document Quantity amount.                                                 |
|             | Original Document Quantity      | Number(12,4) |               | Quantity *10000 (implied 4 decimal places), in EACHES, of the item on this detail record.                                            |
|             | Sign Indicator                  | Char(1)      |               | Indicates either a positive (+) or a negative (-) Original Unit Cost amount.                                                         |
|             | Original Unit cost              | Number(20,4) |               | Unit cost *10000 (implied 4 decimal places), in document currency, of the item on this detail record.                                |
|             | Original VAT Code               | Char (6)     |               | VAT code for item                                                                                                                    |

| Record Name | Field Name                          | Field Type      | Default Value | Description                                                                                                                                             |
|-------------|-------------------------------------|-----------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Original VAT rate                   | Number (20,10)  |               | VAT Rate for the VAT code/item                                                                                                                          |
|             | Sign Indicator                      | Char(1)         |               | Indicates either a positive (+) or a negative (-) total allowance. Default is "+" if no allowances exist for this detail record.                        |
|             | Total Allowance                     | Number(20,4)    |               | Sum of allowance details for this item detail record *10000 (implied 4 decimal places). If no allowances exist for this item detail record, value is 0. |
| TNMRC       | Record descriptor                   | Char(5)         |               | Describes file record type.                                                                                                                             |
|             | Line id                             | Number (10)     |               | Sequential file line number.                                                                                                                            |
|             | Transaction number                  | Number(10)      |               | Transaction number for this non-merchandise record.                                                                                                     |
|             | Non Merchandise Code                | Char(6)         |               | Non-Merchandise code that describes this cost.                                                                                                          |
|             | Sign Indicator                      | Char(1)         |               | Indicates either a positive (+) or a negative (-) Non Merchandise Amt.                                                                                  |
|             | Non Merchandise Amt                 | Number(20,4)    |               | Cost *10000 (implied 4 decimal places) in the document currency.                                                                                        |
|             | Non Merch VAT Code                  | Char (6)        |               | VAT Code for Non-Merchandise                                                                                                                            |
|             | Non Merch Vat Rate at this VAT code | Number (20, 10) |               | VAT Rate corresponding to the VAT code                                                                                                                  |
|             | Service Performed Indicator         | Char(1)         |               | Indicates if a service has actually performed.                                                                                                          |
|             | Store                               | Number(10)      |               | Store at which the service was performed.                                                                                                               |
| TVATS       | File record descriptor              | Char(5)         |               | Marks costs at VAT rate line. Valid value is TVATS.                                                                                                     |
|             | Line id                             | Char(10)        |               | Sequential file line number                                                                                                                             |

| Record Name | Field Name            | Field Type     | Default Value | Description                                                                             |
|-------------|-----------------------|----------------|---------------|-----------------------------------------------------------------------------------------|
|             | Transaction number    | Number(10)     |               | Transaction number for this vat detail record.                                          |
|             | VAT code              | Char(6)        |               | VAT code that applies to cost                                                           |
|             | VAT rate              | Number (20,10) |               | VAT Rate corresponding to the VAT code                                                  |
|             | Sign Indicator        | Char(1)        |               | Indicates either a positive (+) or a negative (-) Original Document Quantity amount.    |
|             | Cost at this VAT code | Number (20,4)  |               | Total amount *10000 (implied 4 decimal places) that must be taxed at the above VAT code |
| TTAIL       | Record descriptor     | Char(5)        |               | Describes file record type. Default value is TTAIL.                                     |
|             | Line id               | Number(10)     |               | Sequential file line number.                                                            |
|             | Transaction number    | Number(10)     |               | Transaction number for the transaction that this record is closing.                     |
|             | Transaction lines     | Number(6)      |               | Total number of detail lines within this transaction.                                   |
| FTAIL       | Record descriptor     | Char(5)        |               | Describes file record type.                                                             |
|             | Line id               | Number(10)     |               | Sequential file line number.                                                            |
|             | Number of lines       | Number(10)     |               | Total number of lines within this file excluding FHEAD and FTAIL.                       |

## invclshp (Invoice Close Shipments)

### Functional Area

Invoice Matching.

### Module Affected

INVCLSHP.PC

### Design Overview

This batch program closes all shipments that have remained open for a specified number of days and are not associated with any open invoices. This is accomplished by setting the invc\_match\_status on the SHIPMENT table to 'C'losed.

**Scheduling Constraints**

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | AD-HOC      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| PERIOD         | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |
| ORDHEAD        | Yes    | No     | No     | No     |
| SHIPMENT       | Yes    | No     | Yes    | No     |
| SHIPSKU        | Yes    | No     | No     | No     |
| INVC_HEAD      | Yes    | No     | No     | No     |
| INVC_XREF      | Yes    | No     | No     | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**invprg (Invoice Purge)****Functional Area**

Invoice Matching

## Module Affected

INVPRG.PC

## Design Overview

This program purges old (older than order\_history\_months on UNIT\_OPTIONS) posted invoices that are not already purged by ORDPRG.PC (which purges invoices associated with an order). This includes all types of invoices—non-merchandise, credit notes, credit note requests, debit memos, and consignment invoices. Regular merchandise invoices are primarily deleted through ORDPRG.PC but are deleted by INVPRG.PC if they still exist in the system. The age of the invoices is determined from the match date; if there is no match date, the invoice date is used.

## Scheduling Constraints

| Schedule Information      | Description                            |
|---------------------------|----------------------------------------|
| Processing Cycle          | AD-HOC (monthly)                       |
| Scheduling Considerations | The program should run after ORDPRG.PC |
| Pre-Processing            | N/A                                    |
| Post-Processing           | N/A                                    |
| Threading Scheme          | N/A                                    |

## Restart/Recovery

N/A

## Locking Strategy

The following tables are row-level locked by shared module INVC\_SQL.DELETE\_INVC for UPDATE:

- ORDLOC\_INVC\_COST

The following tables are row-level locked by the shared module INVC\_SQL.DELETE\_INVC for DELETE:

- INVC\_DETAIL
- INVC\_NON\_MERCH
- INVC\_MERCH\_VAT
- INVC\_DETAIL\_VAT
- INVC\_DISCOUNT
- INVC\_TOLERANCE
- INVC\_HEAD
- INVC\_MATCH\_QUEUE

## Security Considerations

N/A

## Performance Considerations

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| PERIOD           | Yes    | No     | No     | No     |
| UNIT_OPTIONS     | Yes    | No     | No     | No     |
| INVC_HEAD        | Yes    | No     | No     | Yes    |
| SA_TRAN_HEAD     | Yes    | No     | No     | No     |
| SHIPSKU          | Yes    | No     | No     | No     |
| INVC_DETAIL      | No     | No     | No     | Yes    |
| INVC_NON_MECH    | No     | No     | No     | Yes    |
| INVC_MERCH_VAT   | No     | No     | No     | Yes    |
| INVC_DETAIL_VAT  | No     | No     | No     | Yes    |
| INVC_DISCOUNT    | No     | No     | No     | Yes    |
| INVC_TOLERANCE   | No     | No     | No     | Yes    |
| ORDLOC_INVC_COST | No     | No     | Yes    | No     |
| INVC_MATCH_QUEUE | No     | No     | No     | Yes    |

**I/O Specification**

N/A

## Open to Buy Maintenance Batch

### Overview

RMS batch modules provide data to and process data from an external open-to-buy (OTB) planning application. RMS' OTB and stock ledger tables serve as sources of data sent to the planning application, while the planning application returns OTB budget and forward limit figures to RMS.

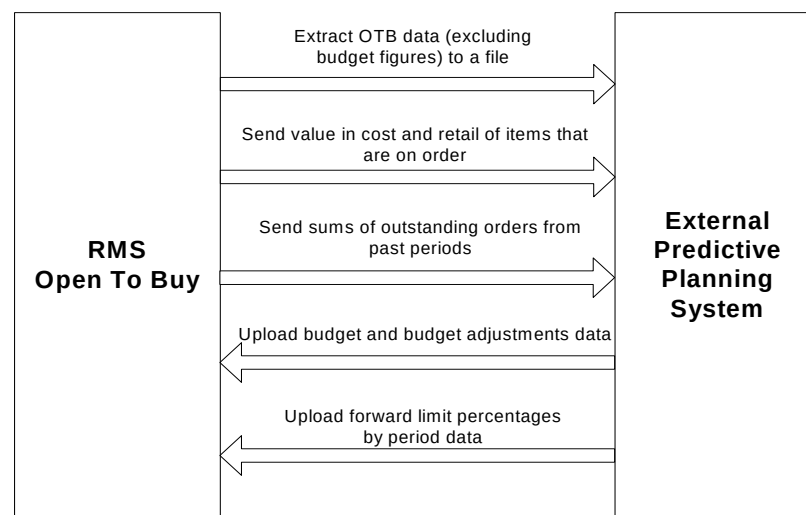
**Note:** RMS 12.0 is **not** integrated with Oracle Retail planning and forecasting systems (for example, Oracle Retail Demand Forecasting).

The customer can choose to send the planning application stock ledger data at the subclass-location-week level either for the most current week or for a historical period. After the planning application processes the RMS data, it returns OTB budget calculations.

This overview describes the RMS batch modules that facilitate the movement of OTB data.

In this version of RMS, the following three part process calculates and exports on order numbers:

1. The on order extract program (ONORDEXT.PC) addresses the ordering tables and breaks out the on order numbers. It inserts these values into ON\_ORDER\_TEMP. These values are at the item/location/week/order\_no level.
2. The on intercompany transfer extract (ONICTEXT.PC) retrieves values items that are crossing between transfer entities on intercompany transfers and inserts the values into ON\_ORDER\_TEMP.
3. The on order download program (ONORDDNLD.PC) retrieves the records on ON\_ORDER\_TEMP, groups them by item/location/week/tsf\_loc\_ind and writes them to three output files to be exported to Oracle Retail Merchandise Financial Planning.



Open To Buy Processes



## Batch Design Summary

The following batch designs are included in this functional area:

- ONICTEXT.PC (On Inter-company Transfer Exhibit)
- ONORDDNLD.PC (On Order Download)
- ONORDEXT.PC (On Order Extract)
- OTBDLORD.PC (Outstanding Order Export Download File)
- OTBDLSAL.PC (Open to Buy Download Stock Ledger)
- OTBDNLD.PC (Open to Buy Download)
- OTBPRG.PC (Open to Buy Purge)
- OTBUFWD.PC (Accept Forward Limit Percentages Upload)
- OTBUPLD.PC (New Budget Data and Budget Adjustments)
- STLGDNLD.PC (Stock Ledger Download)

## onictext (On Inter-company Transfer Exhibit)

### Functional Area

RMS/Planning system interfaces

### Module Affected

ONICTEXT.PC

### Design Overview

This program calculates the value in cost and retail of items that are on intercompany transfers. It calculates the on order cost and retail for all approved intercompany transfers that have exp\_dc\_eow\_dates less than or equal to the planning horizon date. Once the program has calculated the costs and retails, they are inserted into the ON\_ORDER\_TEMP table.

### Scheduling Constraints

| Schedule Information      | Description                         |
|---------------------------|-------------------------------------|
| Processing Cycle          | Phase 4 (Weekly)                    |
| Scheduling Considerations | after onordext and before onorddnld |
| Pre-Processing            | N/A                                 |
| Post-Processing           | N/A                                 |
| Threading Scheme          | Threaded by Transfer number.        |

### Restart/Recovery

The logical unit of work is unique transfer number. Each time the record counter equals the maximum recommended commit number the retek\_commit function is called. The program is multithreaded using v\_restart\_transfer view.

### Locking Strategy

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| STORE            | Yes    | No     | No     | No     |
| WH               | Yes    | No     | No     | No     |
| TSF_ITEM_COST    | Yes    | No     | No     | No     |
| TSFHEAD          | Yes    | No     | No     | No     |
| ITEM_MASTER      | Yes    | No     | No     | No     |
| ITEM_LOC_SOH     | Yes    | No     | No     | No     |
| ITEM_LOC         | Yes    | No     | No     | No     |
| TSF_XFORM        | Yes    | No     | No     | No     |
| TSF_XFORM_DETAIL | Yes    | No     | No     | No     |
| PERIOD           | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS   | Yes    | No     | No     | No     |
| ON_ORDER_TEMP    | No     | Yes    | No     | No     |

**I/O Specification**

## Input File Layout

| Record Name | Field Name                   | Field Type | Default Value | Description                             |
|-------------|------------------------------|------------|---------------|-----------------------------------------|
|             | Weekly or historic indicator | Char (1)   |               | Weekly or historic indicator.           |
|             | Planning horizon start date  | Date (8)   |               | Planning start date in YYYYMMDD format. |
|             | Planning horizon end date    | Date (8)   |               | Planning end date in YYYYMMDD format.   |

**onorddnld (On Order Download to Financial Planning)****Functional Area**

Open to buy maintenance

**Module Affected**

ONORDDNLD.PC

**Design Overview**

This program sends on order cost, retail and quantity at the item/location/week level to a planning system. The values are used by a financial planning system to generate OTB numbers that are interfaced back into the RMS.

This program creates three output files: one for orders, one for intercompany transfer sending locations and one for intercompany transfer receiving locations.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                              |
|---------------------------|----------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 4 (Daily)                                                                                          |
| Scheduling Considerations | This program is run towards end of batch cycle after the ONORDEXT.PC (on order extract) and ONICTEXT.PC. |
| Pre-Processing            | N/A                                                                                                      |
| Post-Processing           | N/A                                                                                                      |
| Threading Scheme          | Threaded by location                                                                                     |

**Restart/Recovery**

The logical unit of work for this program is set at item/location/eow\_date level. Table based restart/recovery must be used. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 1000 records (subject to change based on implementation).

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table         | Select | Insert | Update | Delete |
|---------------|--------|--------|--------|--------|
| ON_ORDER_TEMP | Yes    | No     | No     | No     |

**I/O Specification**

Output File Layout

| Record Name | Field Name                 | Field Type    | Default Value | Description                                                                                                                   |
|-------------|----------------------------|---------------|---------------|-------------------------------------------------------------------------------------------------------------------------------|
|             | ITEM                       | Char(25)      |               | RMS ITEM Identifier                                                                                                           |
|             | Location (Store / WH)      | NUMBER(20)    |               | Store or WH identifier                                                                                                        |
|             | Location Type ('S' or 'W') | Char(1)       |               | Indicates if the location is a store or a warehouse:<br>S – if the location is a store,<br>W – If the location is a warehouse |
|             | OTB EOW date               | DATE (8)      |               | The OTB End of week date.                                                                                                     |
|             | On Order Retail            | NUMBER(25, 4) |               | Total on order retail for the item/location/EOW date.                                                                         |
|             | On order Cost              | NUMBER(25, 4) |               | Total on order cost for the item/location/EOW date.                                                                           |
|             | On Order Quantity          | NUMBER(17, 4) |               | Total on order Quantity for the item/location/EOW date.                                                                       |

## onordext (On Order Extract)

### Functional Area

OTB RMS/Planning interface.

### Module Affected

ONORDEXT.PC

### Design Overview

This program calculates the value in cost and retail of items that are on order for the department/class/subclass/location level. This program is the first step in the stock ledger download process to a planning system. It calculates the on order cost and retail for all approved orders that have not before dates less than or equal to the planning horizon date. Once the program has calculated the costs and retails, they are inserted into the ON\_ORDER\_TEMP table.

### Scheduling Constraints

| Schedule Information      | Description                       |
|---------------------------|-----------------------------------|
| Processing Cycle          | Phase 4 (Daily)                   |
| Scheduling Considerations | before onictext                   |
| Pre-Processing            | Run prepost onordext pre program. |
| Post-Processing           | N/A                               |
| Threading Scheme          | Threaded by Order number          |

## Restart/Recovery

The logical unit of work is unique order number. Each time the record counter equals the maximum recommended commit number the retek\_commit function is called.

It is also split into two sections item and pack. First all items on orders are processed. When they are done a pack 'flag' is turned on and the restart order is reset. Then all the packs on order are processed. All orders are considered twice, once for items and once for packs.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| ORDHEAD                | Yes    | No     | No     | No     |
| ORDLOC                 | Yes    | No     | No     | No     |
| ORDSKU                 | Yes    | No     | No     | No     |
| ALLOC_HEADER           | Yes    | No     | No     | No     |
| ALLOC_DETAIL           | Yes    | No     | No     | No     |
| ITEM_MASTER            | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC  | Yes    | No     | No     | No     |
| PRICE_ZONE_GROUP_STORE | Yes    | No     | No     | No     |
| ITEM_ZONE_PRICE        | Yes    | No     | No     | No     |
| PERIOD                 | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS         | Yes    | No     | No     | No     |
| CLASS                  | Yes    | No     | No     | No     |
| ON_ORDER_TEMP          | No     | Yes    | No     | No     |

## I/O Specification

### Input File Layout:

The input file has only one fixed-length record.

| Record Name | Field Name                   | Field Type | Default Value | Description                                    |
|-------------|------------------------------|------------|---------------|------------------------------------------------|
| N/A         | Weekly or Historic indicator | Char (1)   |               | Weekly or Historic indicator                   |
|             | Start Date                   | Date       |               | Planning horizon start date in YYYYMMDD format |
|             | End Date                     | Date       |               | Planning horizon end date in YYYYMMDD format.  |

## otbdlord (Outstanding Order Export Download)

### Functional Area

OTB – Order to Planning System Interface

### Module Affected

OTBDLORD.PC

### Design Overview

This batch program sums outstanding orders from past periods for each subclass and export the data to a flat file for use by an outside planning system.

Approved orders of each order type are summed by subclass across all periods with an end-of-week date before today. The outstanding order amount is then calculated by subtracting all receipts for each order type by subclass from the approved order amount. This figure is written to the output file for each order type by subclass.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                            |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 4 (Daily).                                                                                                                                                                       |
| Scheduling Considerations | This program must be run after SALWEEK for the week just ended. This program and OTBDLSAL can run anytime after SALWEEK, but SALDLY cannot run between OTBDNLD, OTBDLSAL and OTBDLORD. |
| Pre-Processing            | N/A                                                                                                                                                                                    |
| Post-Processing           | N/A                                                                                                                                                                                    |
| Threading Scheme          | N/A. Table-based array processing is used to speed up performance.                                                                                                                     |

## Restart/Recovery

The logical unit of work for the OTBDLORD module is department/class/subclass. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O. The recommended commit counter setting is 10000 records. Each time the record counter equals the maximum recommended commit number, an application image array record is written to the restart\_start\_array for restart/recovery if a fatal error occurs.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table  | Select | Insert | Update | Delete |
|--------|--------|--------|--------|--------|
| OTB    | Yes    | No     | No     | No     |
| PERIOD | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name                  | Field Type | Default Value       | Description                                                     |
|-------------|-----------------------------|------------|---------------------|-----------------------------------------------------------------|
| File Header | File Type Record Descriptor | Char(5)    | FHEAD               | Identifies file record type                                     |
|             | File Line Sequence Number   | Number(10) | line number in file | Keeps track of the record's position in the file by line number |
|             | File Type Definition        | Char(4)    | OOEX                | Identifies file as 'OTB Outstanding Order Export'               |
|             | File Create Date            | Char(14)   | vdate               | Vdate in YYYYMMDD format .Remaining six characters are blank.   |
| File Detail | File Type Record Descriptor | Char(5)    | FDETL               | Identifies file record type                                     |

| Record Name  | Field Name                     | Field Type | Default Value       | Description                                                                                    |
|--------------|--------------------------------|------------|---------------------|------------------------------------------------------------------------------------------------|
|              | File Line Sequence Number      | Number(10) | line number in file | Keeps track of the record's position in the file by line number                                |
|              | Transaction Set Control Number | Number(14) | sequence number     | Used to force unique detail record check                                                       |
|              | Department                     | Number(4)  |                     | The ID number of a department                                                                  |
|              | Class                          | Number(4)  |                     | The ID number of a class within the department given                                           |
|              | Subclass                       | Number(4)  |                     | The ID number of a subclass within the class given                                             |
|              | N Outstanding Amt              | Number(20) |                     | The amount of outstanding non-basic orders for past periods * 10000 (4 implied decimal places) |
|              | B Outstanding Amt              | Number(20) |                     | The amt. of outstanding buyer-basic orders for past periods* 10000 (4 implied decimal places)  |
| File Trailer | A Outstanding Amt              | Number(20) |                     | The amt. of outstanding auto-basic orders for past periods* 10000 (4 implied decimal places)   |
|              | File Type Record Descriptor    | Char(5)    | FTAIL               | Identifies file record type                                                                    |
|              | File Line Sequence Number      | Number(10) |                     | Keeps track of the record's position in the file by line number                                |
|              | Control Number File Line Count | Number(10) |                     | Total number of all transaction lines, not including file header and trailer.                  |

## otbdlisal (Open To Buy Download Stock Ledger)

### Functional Area

OTB – Stock Ledger to Planning System Interface



**Module Affected**

OTBDLSAL.PC

**Design Overview**

This module sums stock ledger data from the DAILY\_DATA table and opens stock information from the WEEK\_DATA table across the current week, grouping by department, class, subclass, location and date, and export the data to a flat file for use by an outside planning system.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                                                                                                            |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 4 (Weekly)                                                                                                                                                                                                                       |
| Scheduling Considerations | This program must be run after ORDUPD (order upload.) It also must be run after SALWEEK for the week just ended. This program and OTBDNLD can run anytime after SALWEEK, but SALDLY cannot run between OTBDNLD, OTBDLSAL and OTBDLORD. |
| Pre-Processing            | N/A                                                                                                                                                                                                                                    |
| Post-Processing           | N/A                                                                                                                                                                                                                                    |
| Threading Scheme          | N/A. Table-based array processing is used to speed up performance.                                                                                                                                                                     |

**Restart/Recovery**

The logical unit of work for the OTBDLSAL module is department, class, subclass and location. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O. The recommended commit counter setting is 10000 records. Each time the record counter equals the maximum recommended commit number, an application image array record is written to the restart\_start\_array for restart/recovery if a fatal error occurs.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table      | Select | Insert | Update | Delete |
|------------|--------|--------|--------|--------|
| DAILY_DATA | Yes    | No     | No     | No     |
| WEEK_DATA  | Yes    | No     | No     | No     |

| Table  | Select | Insert | Update | Delete |
|--------|--------|--------|--------|--------|
| PERIOD | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name                     | Field Type | Default Value       | Description                                                                                    |
|-------------|--------------------------------|------------|---------------------|------------------------------------------------------------------------------------------------|
| FHEAD       | File Type Record Descriptor    | Char(5)    | FHEAD               | Identifies file record type                                                                    |
|             | File Line Sequence Number      | Number(10) | 0000000001          | Keeps track of the record's position in the file by line number                                |
|             | File Type Definition           | Char(4)    | STKE                | Identifies file as Stock Ledger Export                                                         |
|             | File Create Date               | Char(14)   | vdate               | Date file was written by batch program in YYYYMMDD format. Remaining six characters are blank. |
| FDETL       | File Type Record Descriptor    | Char(5)    | FDETL               | Identifies file record type                                                                    |
|             | File Line Sequence Number      | Number(10) | line number in file | Keeps track of the record's position in the file by line number                                |
|             | Transaction Set Control Number | Number(14) | sequence number     | Used to force unique file check                                                                |
|             | Department                     | Number(4)  |                     | The ID number of a department                                                                  |
|             | Class                          | Number(4)  |                     | The ID number of a class within the department given                                           |
|             | Subclass                       | Number(4)  |                     | The ID number of a subclass within the class given                                             |
|             | Loc_type                       | Char(1)    |                     | The type of the location from which stock ledger data was collected.                           |
|             | Location                       | Number(10) |                     | The location from which stock ledger data was collected.                                       |
|             | Half No.                       | Number(5)  |                     | The half number for this stock ledger data                                                     |

| Record Name | Field Name               | Field Type   | Default Value | Description                                                                                                                        |
|-------------|--------------------------|--------------|---------------|------------------------------------------------------------------------------------------------------------------------------------|
|             | Month No.                | Number(2)    |               | The month number in the half for this stock ledger data                                                                            |
|             | Week No.                 | Number(2)    |               | The week number in the month for this stock ledger data                                                                            |
|             | Open Stock Retail        | Number(20,4) |               | The retail opening stock from the week_data table *10000 (implied 4 decimal places) for this stock ledger period                   |
|             | Open Stock Cost          | Number(20,4) |               | The cost opening stock from the week_data table *10000 (implied 4 decimal places) for this stock ledger period                     |
|             | Stock Adjustments Retail | Number(20,4) |               | The retail stock adjustments summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period       |
|             | Stock Adjustments Cost   | Number(20,4) |               | The cost stock adjustments summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period         |
|             | Purchases Retail         | Number(20,4) |               | The retail purchases summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period               |
|             | Purchases Cost           | Number(20,4) |               | The cost purchases summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period                 |
|             | RTV Retail               | Number(20,4) |               | The retail return to vendor amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period |
|             | RTV Cost                 | Number(20,4) |               | The cost return to vendor amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period   |
|             | Freight Cost             | Number(20,4) |               | The freight cost summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period                   |

| Record Name | Field Name                    | Field Type   | Default Value | Description                                                                                                                          |
|-------------|-------------------------------|--------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------|
|             | Net Sales Retail              | Number(20,4) |               | The retail net sales summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period                 |
|             | Net Sales Cost                | Number(20,4) |               | The cost net sales summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period                   |
|             | Returns Retail                | Number(20,4) |               | The retail returns amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period            |
|             | Returns Cost                  | Number(20,4) |               | The cost returns amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period              |
|             | Promotional Markdowns Retail  | Number(20,4) |               | The retail promotional markdowns summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period     |
|             | Markdown Cancellations Retail | Number(20,4) |               | The retail markdown cancellations summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period    |
|             | Employee Discount Retail      | Number(20,4) |               | The retail employee discounts amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period |
|             | Workroom Amount               | Number(20,4) |               | The workroom amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period                  |
|             | Cash Discount Amount          | Number(20,4) |               | The cash discounts amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period            |
|             | Sales Units                   | Number(12,4) |               | The sales units summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period                      |

| Record Name | Field Name                                    | Field Type   | Default Value | Description                                                                                                                     |
|-------------|-----------------------------------------------|--------------|---------------|---------------------------------------------------------------------------------------------------------------------------------|
|             | Markups Retail                                | Number(20,4) |               | The retail markups summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period              |
|             | Markup Cancellations Retail                   | Number(20,4) |               | The retail markup cancellations summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period |
|             | Clearance Markdowns Retail                    | Number(20,4) |               | The retail clearance markdowns summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period  |
|             | Permanent Markdowns Retail                    | Number(20,4) |               | The retail permanent markdowns summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period  |
|             | Freight Claim Retail                          | Number(20,4) |               | The retail freight claim summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period        |
|             | Freight Claim Cost                            | Number(20,4) |               | The cost freight claim summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period          |
|             | Stock Adjust Cost of Goods Sold (COGS) Retail | Number(20,4) |               | The retail stock adjust COGS summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period    |
|             | Stock Adjust Cost of Goods Sold (COGS) Cost   | Number(20,4) |               | The cost stock adjust COGS summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period      |
|             | Inter-company In Retail                       | Number(20,4) |               | The Inter-company In retail summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period     |
|             | Inter-company In Cost                         | Number(20,4) |               | The Inter-company In cost summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period       |

| Record Name | Field Name                           | Field Type   | Default Value | Description                                                                                                                              |
|-------------|--------------------------------------|--------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------|
|             | Inter-company Out Retail             | Number(20,4) |               | The Inter-company Out Retail summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period             |
|             | Inter-company Out Cost               | Number(20,4) |               | The Inter-company Out Cost summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period               |
|             | Inter-company Markup                 | Number(20,4) |               | The Inter-company Markup summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period                 |
|             | Inter-company Markdown               | Number(20,4) |               | The Inter-company Markdown summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period               |
|             | Work Order Activity Update Inventory | Number(20,4) |               | The Work Order Activity Update Inventory summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period |
|             | Work Order Activity Post Finishing   | Number(20,4) |               | The Work Order Activity Post Finishing summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period   |
| FTAIL       | File Type Record Descriptor          | Char(5)      | FTAIL         | Identifies file record type                                                                                                              |
|             | File Line Sequence Number            | Number(10)   |               | Keeps track of the record's position in the file by line number                                                                          |
|             | Control Number File Line Count       | Number(10)   |               | Total number of all transaction lines, not including file header and trailer                                                             |

## otbdnld (Open To Buy Download)

### Functional Area

OTB-Planning System Interface

**Module Affected**

OTBDNLD.PC

**Design Overview**

This batch program extracts current and future Open to Buy data from the OTB table and export it to a flat file for use by an outside planning system.

All Open to Buy (OTB) data except budget figures is exported directly to the file from the table with no intermediate manipulation.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                                                                                                   |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 4 (Weekly)                                                                                                                                                                                                              |
| Scheduling Considerations | This program must be run after ORDUPD (order upload.) It also must be run after SALWEEK for the week just ended. This program and OTBDLSAL can run anytime after SALWEEK, but SALDLY cannot run between OTBDNLD and OTBDLSAL. |
| Pre-Processing            | N/A                                                                                                                                                                                                                           |
| Post-Processing           | N/A                                                                                                                                                                                                                           |
| Threading Scheme          | N/A. Table-based array processing is used to speed up performance.                                                                                                                                                            |

**Restart/Recovery**

The logical unit of work for the OTBDNLD module is department, class, subclass and end-of-week date with a recommended commit counter setting of 10,000. Each time the record counter equals the maximum recommended commit number, an application image array record is written to the restart\_start\_array for restart/recovery if a fatal error occurs.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table  | Select | Insert | Update | Delete |
|--------|--------|--------|--------|--------|
| OTB    | Yes    | No     | No     | No     |
| PERIOD | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name                     | Field Type  | Default Value | Description                                                              |
|-------------|--------------------------------|-------------|---------------|--------------------------------------------------------------------------|
| FHEAD       | File Type Record Descriptor    | Char (5)    | FHEAD         | Identifies file record type                                              |
|             | File Line Sequence Number      | Number (10) | 0000000001    | Keeps track of the record's position in the file by line number          |
|             | File Type Definition           | Char (4)    | OTBE          | Identifies file as 'OTB Export'                                          |
|             | File Create Date               | Char(14)    |               | Vdate in YYYYMMDD format. Remaining 6 characters are blank.              |
| FDETL       | File record descriptor         | Char(5)     | FDETL         | Identifies file record type                                              |
|             | File Line Sequence Number      | Number (10) |               | Keeps track of the record's position in the file by line number          |
|             | Transaction Set Control Number | Number(14)  |               | Used to force unique file check                                          |
|             | Department                     | Number(4)   |               | The ID number of a department                                            |
|             | Class                          | Number(4)   |               | The ID number of a class within the department given                     |
|             | Subclass                       | Number(4)   |               | The ID number of a subclass within the class given                       |
|             | Eow Date                       | Date        |               | The end of week date for the budgeted period. Format is 'YYYYMMDDHHMMSS' |
|             | Week No.                       | Number(2)   |               | The week number in the month for the budgeted period                     |
|             | Month No.                      | Number(2)   |               | The month number in the half for the budgeted period                     |



| Record Name | Field Name                | Field Type  | Default Value | Description                                                                                                         |
|-------------|---------------------------|-------------|---------------|---------------------------------------------------------------------------------------------------------------------|
|             | Half No.                  | Number(5)   |               | The half number for the budgeted period                                                                             |
|             | Cancel Amt.               | Number(20)  |               | The total amount cancelled from orders of all order type for the budgeted period * 10000 (4 implied decimal places) |
|             | N Approved Amt            | Number(20)  |               | The amount of approved non-basic orders for the budgeted period * 10000 (4 implied decimal places)                  |
|             | N Receipts Amt            | Number(20)  |               | The amount of non-basic orders due in the budgeted period that are received * 10000 (4 implied decimal places)      |
|             | B Approved Amt            | Number(20)  |               | The amount of approved buyer-basic orders for the budgeted period * 10000 (4 implied decimal places)                |
|             | B Receipts Amt            | Number(20)  |               | The amount of buyer-basic orders due in the budgeted period that are received * 10000 (4 implied decimal places)    |
|             | A Approved Amt            | Number(20)  |               | The amount of approved auto-basic orders for the budgeted period * 10000 (4 implied decimal places)                 |
|             | A Receipts Amt            | Number (20) |               | The amount of auto-basic orders due in the budgeted period that are received * 10000 (4 implied decimal places)     |
| FTAIL       | File record descriptor    | Char (5)    | FTAIL         | Identifies file record type                                                                                         |
|             | File Line Sequence Number | Number (10) |               | Keeps track of the record's position in the file by line number                                                     |

| Record Name | Field Name      | Field Type  | Default Value | Description                                                                  |
|-------------|-----------------|-------------|---------------|------------------------------------------------------------------------------|
|             | Number of lines | Number (10) |               | Total number of all transaction lines, not including file header and trailer |

## otbprg (Open to Buy Purge)

### Functional Area

OTB

### Module Affected

OTBPRG.PC

### Design Overview

This batch program runs at the end of the half to delete rows from the OTB table that are at least one half old. The current and previous half's purchase budget data is retained. Open to Buy history can be retained longer with a modification to the function that calculates the purge date.

### Scheduling Constraints

| Schedule Information | Description      |
|----------------------|------------------|
| Processing Cycle     | Ad Hoc (Monthly) |
| Scheduling Diagram   | N/A              |
| Pre-processing       | N/A              |
| Post-Processing      | N/A              |
| Threading Scheme     | N/A              |

### Restart/Recovery

There is no restart/recovery in this module. Up to 10,000 records are deleted and committed at a time to avoid excessive rollback space in usage.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| OTB   | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**otbupfwd (Accept Forward Limit Percentages Upload)****Functional Area**

OTB-Planning System Interface

**Module Affected**

OTBUPFWD.PC

**Design Overview**

The purpose of this batch module is to accept forward limit percentages by period from an outside planning system. This data is then inserted into or used to update the forward limit percentage held on the OTB\_FWD\_LIMIT table.

There is no processing done to these records as the data is transferred directly from the input file to the table. If there is not a row on the table to update, a new row is inserted with the department, class, subclass, period ahead and forward limit percentage as taken from the input file.

**Scheduling Constraints**

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad Hoc      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

**Restart/Recovery**

The logical unit of work for the otbupfwd module is the discrete forward limit transaction. Each record is uniquely identified by department, class, subclass and period ahead. Processing of each row is independent and thus if an erroneous record is found during processing, just that record needs to be corrected and reprocessed.

Error handling is also based on this logical unit of work. If a record fails validation, it is written to a rejected record file. This file facilitates easy reprocessing once the error is fixed by writing the record exactly as it was in the source file.

The recommended commit counter setting is 10,000. If a fatal error occurs and restart is necessary, processing restarts at the last commit point.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table         | Select | Insert | Update | Delete |
|---------------|--------|--------|--------|--------|
| OTB_FWD_LIMIT | No     | Yes    | Yes    | No     |
| SUBCLASS      | Yes    | No     | No     | No     |

**I/O Specification****Input File Layout**

| Record Name | Field Name                     | Field Type | Default Value                | Description                                                                                        |
|-------------|--------------------------------|------------|------------------------------|----------------------------------------------------------------------------------------------------|
| FHEAD       | File head descriptor           | Char(5)    | FHEAD                        | Describes file line type                                                                           |
|             | Line id                        | Number(10) | 0000000001                   | Sequential file line number                                                                        |
|             | File Type Definition           | Char(4)    | FWDI                         | Identifies file as 'FWD Import'                                                                    |
|             | File Create Date               | Char(14)   | <Create date>                | The Date on which the file was written by external system. The Date is in YYYYMMDDHH24MISS format. |
| FDETL       | File record descriptor         | Char(5)    | FDETL                        | Describes file line type                                                                           |
|             | Line ID                        | Number(10) | Line number in file          | Sequential file line number                                                                        |
|             | Transaction Set Control Number | Number(14) | Specified by external system | Used to force unique transaction check                                                             |
|             | Department                     | Number(4)  |                              | The ID number of a department                                                                      |
|             | Class                          | Number(4)  |                              | The ID number of a class within the department given                                               |

| Record Name | Field Name             | Field Type | Default Value       | Description                                                             |
|-------------|------------------------|------------|---------------------|-------------------------------------------------------------------------|
|             | Subclass               | Number(4)  |                     | The ID number of a subclass within the class given                      |
|             | Period Ahead           | Number(2)  |                     | The period ahead (1-99) for the forward limit planning                  |
|             | Forward Limit Percent  | Number(12) |                     | Percentage(0-100) of period's budget available to buyer at current time |
| FTAIL       | File record descriptor | Char(5)    | FTAIL               | Marks end of file                                                       |
|             | Line id                | Number(10) | Line number in file | Sequential file line number                                             |
|             | Number of lines        | Number(10) |                     | Number of lines in file not counting FHEAD and FTAIL                    |

## otbupld (New Budget Data and Budget Adjustments)

### Functional Area

OTB-Planning System Interface

### Module Affected

OTBUPLD.PC

### Design Overview

The purpose of this batch module is to accept new budget data and budget adjustments from an outside planning system. This data is then inserted into or used to update the budget figures on the OTB table.

Detail processing is very limited. If there is not a row on the table to update, a new row is inserted with the department, class, subclass, end-of-week date and budget amount for the specified order type as taken from the input file. Order types tracked by Oracle Retail are Non-Basic (N/B), Buyer Reorder of Basic (BRB) and Automatic Reorder of Basic (ARB). This module does not use Direct Store Delivery (DSD) orders because they do not have records on the OTB table. The date procedures CAL\_TO\_454 and CAL\_TO\_454\_HALF are used to calculate the half, month and week numbers of the record's end-of-week date, which is also held on the OTB table. If a budget amount is being adjusted, the old budget amount for the specified department, class, subclass, end-of-week date and order type is overwritten with the budget amount extracted from the input file.

## Scheduling Constraints

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad Hoc      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

## Restart/Recovery

The logical unit of work for the OTBUPLD module is the discrete budget transaction identified by order type, department, class, subclass and end-of-week date. Processing of each row is independent and thus if an erroneous record is found during processing, only that record needs to be corrected and reprocessed.

Error handling also is based on this logical unit of work. If a record fails validation, it is written to a rejected record file. This file facilitates easy reprocessing once the error is fixed by writing the record exactly as it was in the source file.

The recommended commit counter setting is 10,000. If a fatal error occurs and restart is necessary, processing restarts at the last commit point.

## Locking Strategy

The OTBUPLD module does not lock any table directly but the shared module OTB\_SQL.OTB\_INSERTS, which actually does the upload into OTB table, first locks the matching rows in OTB table for update and then does the actual update/insert.

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| OTB   | No     | Yes    | Yes    | No     |

## I/O Specification

### Input File Layout

| Record Name | Field Name           | Field Type | Default Value | Description                 |
|-------------|----------------------|------------|---------------|-----------------------------|
| FHEAD       | File head descriptor | Char(5)    | FHEAD         | Describes file line type    |
|             | Line id              | Number(10) | 0000000001    | Sequential file line number |

| Record Name | Field Name                     | Field Type | Default Value                | Description                                                                                                                       |
|-------------|--------------------------------|------------|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
|             | File Type Definition           | Char(4)    | 'OTBI'                       | Identifies file as 'OTB Import'                                                                                                   |
|             | File Create Date               | Char(14)   | <Create date>                | The date on which the file was written by external system. The Date is in YYYYMMDDHH24MISS format.                                |
| FDETL       | File record descriptor         | Char(5)    | FDETL                        | Describes file line type                                                                                                          |
|             | Line ID                        | Number(10) | Line number in file          | Sequential file line number                                                                                                       |
|             | Transaction Set Control Number | Number(14) | Specified by external system | Used to force unique transaction check                                                                                            |
|             | Order Type                     | Char(1)    | A, B or N                    | Order type budgeted for: specified as A for auto-basic replenishment, B for buyer-basic replenishment, and N for non-basic orders |
|             | Department                     | Number(4)  |                              | The ID number of a department                                                                                                     |
|             | Class                          | Number(4)  |                              | The ID number of a class within the department given                                                                              |
|             | Subclass                       | Number(4)  |                              | The ID number of a subclass within the class given                                                                                |
|             | Eow Date                       | Char(14)   |                              | The end of week date for the budgeted week in YYYYMMDDHH24MISS format.                                                            |
|             | Budget Amount                  | Number(20) |                              | Budgeted amount for the specified order type/week * 10000 (4 implied decimal places)                                              |
| FTAIL       | File record descriptor         | Char(5)    |                              | Marks end of file                                                                                                                 |
|             | Line ID                        | Number(10) | Line number in file          | Sequential file line number                                                                                                       |

| Record Name | Field Name      | Field Type | Default Value      | Description                                          |
|-------------|-----------------|------------|--------------------|------------------------------------------------------|
|             | Number of lines | Number(10) | Total detail lines | Number of lines in file not counting FHEAD and FTAIL |

## stlgdnld (Stock Ledger Download)

### Functional Area

Finance –Stock Ledger

### Module Affected

STLGDNLD.PC

### Design Overview

This program extracts stock ledger data at the item level. The program can extract data for a historic period or for the most current complete week. The program accepts an input file that determines whether the extract is a historic extract or a weekly extract.

### Scheduling Constraints

| Schedule Information      | Description                |
|---------------------------|----------------------------|
| Processing Cycle          | Phase 4 (Weekly) or ad hoc |
| Scheduling Considerations | N/A                        |
| Pre-Processing            | N/A                        |
| Post-Processing           | N/A                        |
| Threading Scheme          | Multi-threaded by dept     |

### Restart/Recovery

The logical unit of work for this program is set at item, location type, location and date. Threading is done by dept using the v\_restart\_dept view to thread properly.

The changes are posted when the commit\_max\_ctr value is reached. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The value of the counter is subject to change based on implementation.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A



**Key Tables Affected**

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| TRAN_DATA_HISTORY | Yes    | No     | No     | No     |

**I/O Specification****Output File Layout**

| Record Name | Field Name           | Field Type   | Default Value | Description                                       |
|-------------|----------------------|--------------|---------------|---------------------------------------------------|
|             | Item                 | Char(25)     |               | Item number                                       |
|             | Location Type        | Char(1)      |               | Location Type<br>Valid values are 'S','W'         |
|             | Location             | Number(20)   |               | Location Number                                   |
|             | Eow_date             | Char(8)      |               | End of Week date in 'YYYYMMDD' format             |
|             | Update_Ind           | Char(1)      |               | Update Indicator<br>Valid values are 'I ' and 'U' |
|             | Regular_sales_retail | Number(25,4) |               | Regular sales value (retail)                      |
|             | Regular_sales_cost   | Number(25,4) |               | Regular sales value (cost)                        |
|             | Regular_sales_units  | Number(17,4) |               | Regular sales value (units)                       |
|             | Promo_sales_retail   | Number(25,4) |               | Promo sales value (retail)                        |
|             | Promo_sales_cost     | Number(25,4) |               | Promo sales value (cost)                          |
|             | Promo_sales_units    | Number(17,4) |               | Promo sales value (units)                         |
|             | Clear_sales_retail   | Number(25,4) |               | Clearance sales value (retail)                    |
|             | Clear_sales_cost     | Number(25,4) |               | Clearance sales value (cost)                      |
|             | Clear_sales_units    | Number(17,4) |               | Clearance sales value (units)                     |

| Record Name | Field Name                 | Field Type   | Default Value | Description                        |
|-------------|----------------------------|--------------|---------------|------------------------------------|
|             | Sales_retail_excluding_vat | Number(25,4) |               | Sales value excluding vat (retail) |
|             | Custom_returns_retail      | Number(25,4) |               | Custom returns value (retail)      |
|             | Custom_returns_cost        | Number(25,4) |               | Custom returns value (cost)        |
|             | Custom_returns_units       | Number(17,4) |               | Custom returns value (units)       |
|             | Rtv_retail                 | Number(25,4) |               | Return to Vendor value (retail)    |
|             | Rtv_cost                   | Number(25,4) |               | Return to Vendor value (cost)      |
|             | Rtv_units                  | Number(17,4) |               | Return to Vendor value (units)     |
|             | Reclass_in_retail          | Number(25,4) |               | Reclass In value (retail)          |
|             | Reclass_in_cost            | Number(25,4) |               | Reclass In value (cost)            |
|             | Reclass_in_units           | Number(17,4) |               | Reclass In value (units)           |
|             | Reclass_out_retail         | Number(25,4) |               | Reclass Out value (retail)         |
|             | Reclass_out_cost           | Number(25,4) |               | Reclass Out value (cost)           |
|             | Reclass_out_units          | Number(17,4) |               | Reclass Out value (units)          |
|             | Perm_markdown_value        | Number(25,4) |               | Permanent markdown value (retail)  |
|             | Prom_markdown_value        | Number(25,4) |               | Promotion markdown value (retail)  |
|             | Clear_markdown_value       | Number(25,4) |               | Clearance markdown value (retail)  |
|             | Markdown_cancel_value      | Number(25,4) |               | Markdown cancel value              |
|             | Markup_value               | Number(25,4) |               | Markup value                       |
|             | Markup_cancel_value        | Number(25,4) |               | Markup cancel value                |
|             | Stock_adj_retail           | Number(25,4) |               | Stock adjustment value (retail)    |
|             | Stock_adj_cost             | Number(25,4) |               | Stock adjustment value (cost)      |

| Record Name | Field Name              | Field Type   | Default Value | Description                    |
|-------------|-------------------------|--------------|---------------|--------------------------------|
|             | Stock_adj_units         | Number(17,4) |               | Stock adjustment value (units) |
|             | Received_retail         | Number(25,4) |               | Received value (retail)        |
|             | Received_cost           | Number(25,4) |               | Received value (cost)          |
|             | Received_units          | Number(17,4) |               | Received value (units)         |
|             | Tsf_in_retail           | Number(25,4) |               | Transfer In value (retail)     |
|             | Tsf_in_cost             | Number(25,4) |               | Transfer In value (cost)       |
|             | Tsf_in_units            | Number(17,4) |               | Transfer In value (units)      |
|             | Tsf_out_retail          | Number(25,4) |               | Transfer Out value (retail)    |
|             | Tsf_out_cost            | Number(25,4) |               | Transfer Out value (cost)      |
|             | Tsf_out_units           | Number(17,4) |               | Transfer Out value (units)     |
|             | Freight_cost            | Number(25,4) |               | Freight cost                   |
|             | Employee_disc_retail    | Number(25,4) |               | Employee disc (retail)         |
|             | Cost_variance           | Number(25,4) |               | Cost variance                  |
|             | Wkroom_other_cost_sales | Number(25,4) |               | Wkroom other sales (cost)      |
|             | Cash_disc_retail        | Number(25,4) |               | Cash disc (retail)             |
|             | Freight_claim_retail    | Number(25,4) |               | Freight Claim (retail)         |
|             | Freight_claim_cost      | Number(25,4) |               | Freight Claim (cost)           |
|             | Freight_claim_units     | Number(25,4) |               | Freight Claim (Units)          |
|             | Stock_adj_cogs_retail   | Number(25,4) |               | Stock Adjust COGS (retail)     |
|             | Stock_adj_cogs_cost     | Number(25,4) |               | Stock Adjust COGS (cost)       |
|             | Stock_adj_cogs_units    | Number(25,4) |               | Stock Adjust COGS (Units)      |
|             | Intercompany_in_retail  | Number(25,4) |               | Intercompany In value (retail) |
|             | Intercompany_in_cost    | Number(25,4) |               | Intercompany In value (cost)   |

| Record Name | Field Name                  | Field Type   | Default Value | Description                                       |
|-------------|-----------------------------|--------------|---------------|---------------------------------------------------|
|             | Intercompany_in_units       | Number(25,4) |               | Intercompany In value (units)                     |
|             | Intercompany_out_retail     | Number(25,4) |               | Intercompany Out value (retail)                   |
|             | Intercompany_out_cost       | Number(25,4) |               | Intercompany Out value (cost)                     |
|             | Intercompany_out_units      | Number(25,4) |               | Intercompany Out value (units)                    |
|             | Intercompany_markup         | Number(25,4) |               | Intercompany Markup                               |
|             | Intercompany_markup_units   | Number(25,4) |               | Intercompany Markup (units)                       |
|             | Intercompany_markdown       | Number(25,4) |               | Intercompany Markdown                             |
|             | Intercompany_markdown_units | Number(25,4) |               | Intercompany Markdown (units)                     |
|             | Wo_activity_upd_inv         | Number(25,4) |               | Work Order Activity – Update Inventory (cost)     |
|             | Wo_activity_upd_inv_units   | Number(25,4) |               | Work Order Activity – Update Inventory (units)    |
|             | Wo_activity_post_fin        | Number(25,4) |               | Work Order Activity – Post to Financials (retail) |
|             | Wo_activity_post_fin_units  | Number(25,4) |               | Work Order Activity – Post to Financials (units)  |

---

## Oracle Retail Sales Audit Batch

### Overview

Oracle Retail Sales Audit (ReSA) is another member of the Oracle Retail suite of products. The purpose of ReSA is to accept transaction data from point-of-sale (POS) applications and move the data through a series of processes that culminate in “clean” data. This “clean” data can be accepted by a number of systems, including consumer (customer) and merchandising applications. Data that ReSA finds to be inaccurate is brought to the attention of the retailer’s sales auditors who can use the features of the sales audit system to correct the exceptions.

By using ReSA, retailers can quickly and accurately validate and audit transaction data before it is exported to other applications. ReSA uses several batch-processing modules to:

- Import POS transaction data sent from the store to the ReSA database
- Produce totals from user-defined totaling calculation rules that a user can review during the interactive audit
- Validate transaction and total data with user-defined audit rules that generate errors whenever data does not meet the criteria. The user can review these errors during the interactive audit
- Create and export files in formats suitable for transfer to other applications
- Update the ReSA database with adjustments received from external systems on previously exported data

This document describes these processes and the batch processing modules involved with them.

---

**Note:** Oracle Retail Sales Audit is only compatible with the current version of RMS and cannot be used with previous versions of RMS.

---

### Store Day Defined

The term **store day** is used throughout this document. Store day describes all transactions that occur in one business day at one store or location. Because retailers need the ability to audit transactions on a store-by-store basis for a defined period of time, store day data is maintained separately beginning with the initial import of data from the POS system.

### Making Changes in the CODE\_DETAIL Table

After making changes in the code\_detail table for code\_types that ReSA uses, the library programs must be recompiled. Follow these steps:

1. Go to the \$l directory and recompile ‘libresa.a’ and ‘install’:  
make -f retek.mk resa  
make -f retek.mk install

2. Go to the \$c directory and recompile the next libraries:
 

```
make -f mts.mk resa-libchange
make -f mts.mk resa
```

  - a. Recompile the appropriate library depending upon which of the following products is being used:
    - resa-rms
    - resa-rdw
    - resa-ach
    - resa-uar
    - resa-im

```
make -f mts.mk ( name of library )
```
  - b. 

```
make -f mts.mk resa-install
```

## Preparation for the Data Import

The first two batch modules run prior to the importing and processing of the transaction log(s) for a store day.

The batch module SASTDYCR.PC runs to prepare the database tables with information on data that is expected for import and export for each store on the following day. The module looks for all stores that are scheduled to be open the next day, that is, those for which ReSA expects to receive a transaction log. The module then creates a store day record in the ReSA store day table record for that business day and also creates records in the import and export log tables, as well as in the flash sales tables. (A flash sales report shows sales for any time during the day.) SASTDYCR.PC also assures that no duplicate import and export records are created for a store day.

SAGETREF.PC retrieves the following data from RMS and ReSA databases and writes it to the following temporary reference files:

| SAGETREF'S TEMPORARY REFERENCE FILES |                                                                          |
|--------------------------------------|--------------------------------------------------------------------------|
| itemfile                             | transaction level items                                                  |
|                                      | merchandise hierarchy                                                    |
|                                      | standard units of measure (UOM)                                          |
| wastefile                            | transaction level items                                                  |
|                                      | wastage types                                                            |
|                                      | wastage percentages                                                      |
| refitemfile                          | sub-transaction level items and their associated transaction level items |
| primvariantfile                      | locations                                                                |
|                                      | parent items                                                             |
|                                      | primary variants (transaction level)                                     |
| varupcfile                           | information for 'decoding' variable weight PLUs                          |
| storedayfile                         | locations                                                                |
|                                      | dates                                                                    |

| SAGETREF'S TEMPORARY REFERENCE FILES |                                                                              |
|--------------------------------------|------------------------------------------------------------------------------|
|                                      | system codes                                                                 |
|                                      | decimal points in local currency                                             |
| promfile                             | promotions                                                                   |
|                                      | status                                                                       |
| codesfile                            | all codes in the system                                                      |
| errorfile                            | all Sales Audit specific error codes                                         |
| ccvalfile                            | credit card details used for validation                                      |
| storeposfile                         | POS starting and ending transaction numbers for each store                   |
| tendertypefile                       | valid tender type information (credit cards, traveler's checks, cash...)     |
| merchcodesfile                       | non-merchandise codes (snow shoveling, for example)                          |
| partnerfile                          | partner information (banks, agents, etc; includes everyone except suppliers) |
| supplierfile                         | suppliers                                                                    |
|                                      | status                                                                       |
| employeefile                         | store/employee/POS id relationships                                          |
| bannerfile                           | banner IDs                                                                   |

The transaction import module accesses these files when it validates store day data. Being able to read from reference files, instead of from the database itself, speeds the import and validation process. Performance is boosted because interaction with the database is limited.

When running SAGETREF.PC, retailers can either create and specify the output files, or create only the output that they desire. For example, a retailer interested in only creating a more recent employeefile would simply place a “-” in place of all the other parameters but still specify an employeefile name. This technique can be applied to as many or as few of the parameters as retailers wish. Note, however, that the item-related files (itemfile, refitemfile, wastefile, and primvariantfile) contain significant interdependence. Thus, item files must all be created or *not* created together.

## ReSA's Conversion from the Selling UOM to the Standard UOM

In the list of SAGETREF.PC's output files above, “standard UOM” is part of the itemfile. To obtain the value, ReSA converts the selling UOM to the standard UOM during batch processing. This conversion enables ReSA to export the standard UOM to the systems that require its use.

For example, the selling unit of measure is used by RMS to set up retail, promotional, and clearance pricing at the price zone/location level. The selling UOM is downloaded to the POS after the selling units of measure are converted to standard units of measure for reporting purposes. RMS uses the standard UOM to track an item's performance internally in RMS. The standard UOM is used with purchase orders, stock, inventory, sales history, and forecasting.

## A Note about Primary Variant Relationships

Depending upon a retailer's system parameters, the retailer designates the primary variant during item setup (through the front end) for several reasons. One of the reasons is that, in some cases, an item may be identified at the POS by the item parent, but the item parent may have several variants.

The primary variant is established through a form at the item location level. The retailer designates which variant item is the primary variant for the current transaction level item. For more information about the new item structure in RMS, see the Oracle Retail Merchandising System User Guide.

In the example shown in the diagram below, the retailer has established their transaction level as an Item Level 2. Note that the level of the primary variant is Item Level 1, and Item Level 3 is the sub-transaction level (the refitem).

The retailer set up 'golf shirts' in the merchandising system as its Item Level 1 above the transaction level. The retailer set up two items at level 2 (the transaction level) based on size (small and medium). Note that the retailer assigned the level 2 items to all of the available locations (Minneapolis, China, and Fargo). The retailer also designated a primary variant for a single location – a medium golf shirt, in the case of Minneapolis, and a small golf shirt, in the case of China. The retailer failed to designate a primary variant for Fargo.

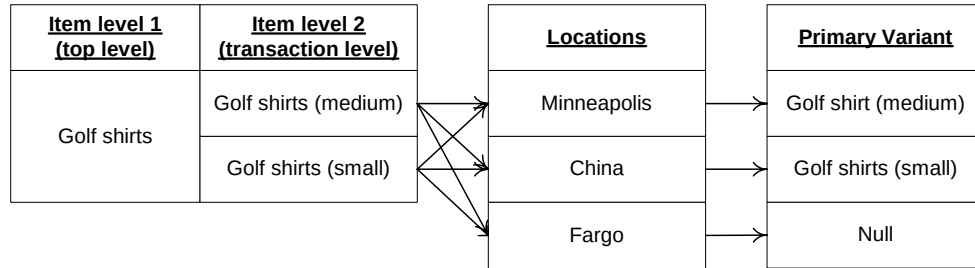
The primary variant affects ReSA in the following way. Sometimes a POS system does not provide ReSA with item level 2 (transaction item) data. For example, assume that the POS system in Minneapolis sold 10 medium golf shirts and 10 small golf shirts but only informed ReSA that 20 golf shirts were sold. '20 golf shirts' presents a problem for ReSA because it can only interpret items at item level 2 (the transaction level). Thus, because 'medium golf shirts' was the chosen primary variant for Minneapolis, the SAGETREF.PC module automatically transforms the '20 golf shirts' into '20 medium golf shirts'. If the same type of POS system in China informed ReSA of '20 golf shirts' (instead of the 10 medium and 10 small that were sold), the SAGETREF.PC module would transform the '20 golf shirts' sold in China into '20 small golf shirts'. As the table shows, 'small golf shirts' was the chosen primary variant for the China location. ReSA then goes on to export the data at the item 2 level (the transaction level) to, for example, a merchandising system, a data warehouse, and so on.

---

**Note:** Depending upon system parameters, if a retailer fails to set up the primary variant for a location, an 'invalid item error' is generated during batch processing. In the example below, if the POS system in Fargo sold 10 medium golf shirts and 10 small golf shirts, but only informed ReSA that 20 golf shirts were sold, the SAGETREF.PC module would not have a way to transform those 20 golf shirts to the transaction level. Because ReSA can only interpret items above the transaction level in conjunction with a primary variant, the 'invalid item error' would occur during batch processing.

---





Primary variant relationships

## Transaction Data Import and Validation

SAIMPTLOG.PC and SAIMPTLOGFIN.PC perform the following:

- Import the transaction log from the POS
- Lock the store day records
- Validate transactions
- Check balances for over or under
- Verify the end of the store day's received transactions
- Unlock the store day records if all transactions for the day are imported

Before SAIMPTLOG.PC can begin to process a store day's transactions, it must receive a transaction log from the retailer's POS that is in an Oracle Retail compatible file format, called the RTLOG (published in this Operations Guide). The retailer is responsible for converting its transaction logs to RTLOGs.

1. SAIMPTLOG.PC locks the store day records in the ReSA database and begins to validate the transaction data against the SAGETREF.PC output. That output is described in the "Preparation for the data import" section earlier in this chapter.
2. SAIMPTLOG.PC looks for duplicate or missing transaction numbers.
3. Errors in transactions are written to error tables.
4. SAIMPTLOG.PC looks for transactions that involve a voucher (gift certificates issued or redeemed or other credit vouchers). It writes those voucher transactions to a file for later processing by SAVOUCH.PC.
5. SAIMPTLOG.PC produces output files that are loaded into ReSA's database, using the Oracle SQL\*Load tool. SQL\*Load is another timesaving technique that speeds the batch process.

## Trickle Polling

ReSA also contains SAIMPTLOGI.PC, which can be used in lieu of SAIMPTLOG.PC. SAIMPTLOGI.PC performs the same functions as SAIMPTLOG.PC but its output is directly inserted into the applicable ReSA table, rather than to a flat file loaded with the Oracle SQL\*Load tool. A retailer trickle polling or exporting a relatively small TLOG would be a good candidate to use SAIMPTLOGI.PC.

| ReSA Valid Transaction Types |                  |
|------------------------------|------------------|
| Transaction Code             | Transaction Type |
| OPEN                         | Open             |

| ReSA Valid Transaction Types |                                                                                                                                                                                                     |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CLOSE                        | Close                                                                                                                                                                                               |
| COND                         | Daily Store Conditions                                                                                                                                                                              |
| DCLOSE                       | Day close indicator                                                                                                                                                                                 |
| LOAN                         | Loan                                                                                                                                                                                                |
| METER                        | Meter Reading for Fuel                                                                                                                                                                              |
| NOSALE                       | No Sale                                                                                                                                                                                             |
| PAIDIN                       | Paid In                                                                                                                                                                                             |
| PAIDOU                       | Paid Out                                                                                                                                                                                            |
| PULL                         | Pull                                                                                                                                                                                                |
| PUMPT                        | Pump Test for Fuel                                                                                                                                                                                  |
| PVOID                        | Post Void (A transaction that was rung later into the register to void something that occurred earlier at the same store/day. A post void updates the original transaction's sub-transaction type.) |
| REFUND                       | Return of customer's original check.                                                                                                                                                                |
| RETURN                       | Return                                                                                                                                                                                              |
| SALE                         | Sale                                                                                                                                                                                                |
| TANKDP                       | Tank Dip                                                                                                                                                                                            |
| TOTAL                        | POS generated totals                                                                                                                                                                                |
| EEXCH                        | Even exchange                                                                                                                                                                                       |
| VOID                         | Void (aborted transaction)                                                                                                                                                                          |

## The DCLOSE Transaction Type

When the retailer is sending only one file to the system, SAIMPTLOG.PC marks the store day record in the ReSA import log as partially or fully loaded in the database by looking for a transaction type of DCLOSE. However, if the retailer is sending more than one file (as in, for example, a trickle polling situation), the retailer can specify the number of files that the system should expect in combination with the DCLOSE transaction type. This ensures that the system receives all of the files, even if the DCLOSE transaction type is, for some reason, received before the final file.

For example, if 24 files are expected over a given amount of time, and the file with the DCLOSE transaction type is, for some reason, sent before the 24<sup>th</sup> file, the RMS system waits until the last file arrives before marking the store day record as partially or fully loaded in the database.

The import process is completed after SAIMPTLOGFIN.PC has updated the store, data and audit status of each store day record.

## Total Calculations and Rules

By providing additional values against which auditors can compare receipts, totaling is integral to the auditing process. Totaling also provides quick access to other numeric figures about the day's transactions.

Totaling in ReSA is dynamic. ReSA automatically totals transactions based on calculation definitions that the retailer's users create using the online Totals Calculation Definition Wizard. In addition, the retailer is able to define totals that come from the POS but that ReSA does not calculate. Whenever users create new calculation definitions or edit existing ones, they become part of the automated totaling process the next time that SATOTALS.PC runs.

Evaluating rules is also integral to the auditing process. Rules make the comparisons among data from various sources. These comparisons find data errors that could be the result of either honest mistakes or fraud. Finding these mistakes during the auditing process prevents these errors from being passed on to other systems, (for example, a merchandising system, a data warehouse system, and so on).

Like totaling, rules in ReSA are dynamic. They are not predefined in the system—retailers have the ability to define them through the online Rules Calculation Definition Wizard.

Errors uncovered by these rules are available for review during the interactive audit. Like SATOTALS.PC, after users modify existing rules or create new ones, they become part of the rules the next time that SARULES.PC runs.

## Export Store Day Transaction Data to Applications

ReSA can prepare data for export to applications after:

- Some or all of the transactions for the day are imported (depending upon the application receiving ReSA's export)
- Totals have run
- Audit rules have run
- Errors in transactions and totals relevant for the system receiving the associated data is eliminated or overridden

ReSA uses separate batch modules to process export data to the external applications described in the table below. Depending upon the application, exported data consists of either transaction data or totals, or both. The table shows you the name of the application to which ReSA exports data, a description of the kind of data processed, and the ReSA batch module that processes data for that application:

**Note:** All of the applications listed in the following table have flat file specifications. See the appropriate design documents for details of those specifications.

| Application Name                         | Data Exported                                   | ReSA Batch Module Name |
|------------------------------------------|-------------------------------------------------|------------------------|
| Oracle Retail Merchandising System (RMS) | Sales, return, exchange, and so on transactions | SAEXPRMS.PC            |

| <b>Application Name</b>               | <b>Data Exported</b>                                                                                                                                                                                                                                                                                                                                                                                                                                       | <b>ReSA Batch Module Name</b>                                                           |
|---------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|
| Oracle Retail Data Warehouse (RDW)    | Transaction item details for: <ul style="list-style-type: none"> <li>• All sales</li> <li>• Returns</li> <li>• Exchanges</li> <li>• Even exchanges</li> <li>• Paid-ins</li> <li>• Paid-outs</li> <li>• No-sales</li> <li>• Voids</li> <li>• Post voids</li> <li>• Store conditions (weather, traffic, temp, and so on)</li> <li>• Transaction tender details</li> <li>• Store-level totals</li> <li>• Cashier or register over or short totals.</li> </ul> | SAEXPRDW.PC                                                                             |
| Account Clearing House (ACH)          | Bank deposit totals<br>Store/day deposit totals                                                                                                                                                                                                                                                                                                                                                                                                            | SAEXPACH.PC                                                                             |
| Reconciliation System (UAR-Driscoll)  | Totals for: <ul style="list-style-type: none"> <li>• Lottery sales</li> <li>• Bank deposits</li> <li>• Money order totals</li> <li>• Credit card totals</li> </ul>                                                                                                                                                                                                                                                                                         | SAEXPUAR.PC                                                                             |
| Oracle Retail Invoice Matching (ReIM) | Invoice number<br>Vendor number<br>Payment reference number<br>Proof of delivery number<br>Payment date<br>Paid indicator                                                                                                                                                                                                                                                                                                                                  | SAEXPIM.PC                                                                              |
| Oracle Retail Invoice Matching (ReIM) | Escheatment totals for each state or country as defined by the retailer                                                                                                                                                                                                                                                                                                                                                                                    | SAESCHEAT.PC writes records for this data to tables that are read into ReIM by SAEXPIM. |

## Transaction Data Exports and the Unit of Work

The process of exporting transaction data varies according to the unit of work selected in ReSA's system options. There are two units of work, transaction and store day. If the unit of work selection is transaction, ReSA exports transactions to RMS as soon as they are free of errors. If the unit of work selection is store day, transactions are not exported until all errors for that store day are either overridden or corrected.

## Oracle Retail Merchandise System (RMS) Export

SAEXPRMS.PC transfers store day transaction data to RMS and rolls up transaction data to the item/store/day/pricepoint level. In other words, RMS receives one sum total of items sold at a particular pricepoint. The pricepoint is item/location/price/dropship. The drop\_ship indicator indicates whether the item is being sent from the location's inventory or directly from the vendor to the customer.

It then writes the data to a file called POSU. This file is available for upload by RMS' POSUPLD.PC batch module. If a ReSA user later modifies any transactions after the store day is exported, SAEXPRMS.PC notes the flagged changes and re-exports that data to RMS. See the section, 'Full disclosure and post-export changes', later in this chapter.

## Oracle Retail Data Warehouse (RDW) export

---

**Note:** ReSA currently contains logic that allows it to export only to the most current version of RDW.

---

SAEXPRDW.PC writes one output file for each for the following:

- Transaction item data
- Transaction tender data
- Store total data
- Cashier or register total data.

Each of these files is then made available to the RDW batch module responsible for uploading the data into the data warehouse. If a ReSA user later modifies any transactions or totals after the store day is exported, SAEXPRDW.PC notes the flagged changes and re-exports that data to RDW. See the section, 'Full disclosure and post-export changes', later in this chapter.

## Account Clearing House (ACH) Export

SAEXPACH.PC produces anticipatory deposit totals for ACH processing. The next business day's deposit is estimated based upon the average of deposits for the same business day of the week for the past four weeks. The current day's actual deposit is compared to the estimated amount from the previous day, and the difference is added or subtracted from the estimated amount for the next day. SAEXPACH.PC formats deposit amounts to a standard BAI version 2 file for export to ACH. BAI is the Bank Administration Institute. Note that the ACH export deviates from the typical Sales Audit export in that store/days must be exported by estimate even though errors may have occurred for a given day or store (depending on the unit of work defined). SAEXPACH.PC functions under the assumption that there is only one total to be exported for ACH processing per store/day.

## Universal Account Reconciliation System (UAR) Export

SAEXPUAR.PC selects lottery, bank deposit, money order, and credit card totals and writes them to output files for export to the J. Driscoll & Associates' UAR application. For each store day, SAEXPUAR.PC posts all specified totals to their appropriate output files.

**Note:** Defining Totals for ACH, and UAR Retailers need to define totals to be exported to ACH and UAR using the online wizards. Totals described here are only examples of those that a retailer might choose to define and later export.

### Oracle Retail Invoice Matching (ReIM) Export

For retailers that have ReIM, SAEXPIM provides invoicing support for Direct Store Delivery (DSD) by transferring transactions for invoices paid out at the store (that are imported by ReSA from the POS) to ReIM. ReIM then uses that data to create an invoice for DSD. Data exported to ReIM by this batch module includes:

- Invoice number
- Vendor number
- Payment reference number
- Proof of delivery number
- Payment date
- Paid indicator

### Escheatment Totals to ReIM for Accounts Payable

The laws of individual states and countries require a retailer to return monies for aged, unclaimed gift certificates and vouchers. This process is called 'escheatment'. SAESCHEAT.PC writes records for this data to tables that are read into ReIM by SAEXPIM.PC. The data can then be sent as invoices approved for payment to a financial application.

## Full Disclosure and Post-export Changes

If a retailer modifies data during the interactive audit that was previously exported to RMS or RDW, ReSA export batch modules re-export the modified data in accordance with a process called *full disclosure*. Full disclosure means that any previously exported values (dollars, units, and so on) are fully backed out before the new value is sent. Here is an example. Suppose that a transaction originally shows a sale of 12 items and that this transaction is exported. Later, during the interactive audit, a retailer determines that the correct amount is 15 items (three more than the original amount) and makes the change. ReSA then flags the corrected amount for export to the application.

Now during the export process, instead of simply adding three items to that transaction (which would change the amount from 12 to 15), a minus 12 (-12) is sent to back out the original amount of 12. Then an amount of 15 is sent. The result is that a transaction is corrected by fully accounting for the original amount before adding the correct one. Full disclosure, then, is meant to completely account for all adjustments.

## What Happens to Totals when Transactions are Modified?

If a retailer modifies transactions during the ReSA interactive audit process, the totaling and auditing processes run again to recalculate store day totals. The batch module SAPREEXP.PC tracks all changed totals for the store day since the last export by comparing the latest prioritized version of each total defined for export with the version that was previously sent to each system. The module writes the changes to revision tables that the export modules later recognize as ready for export.

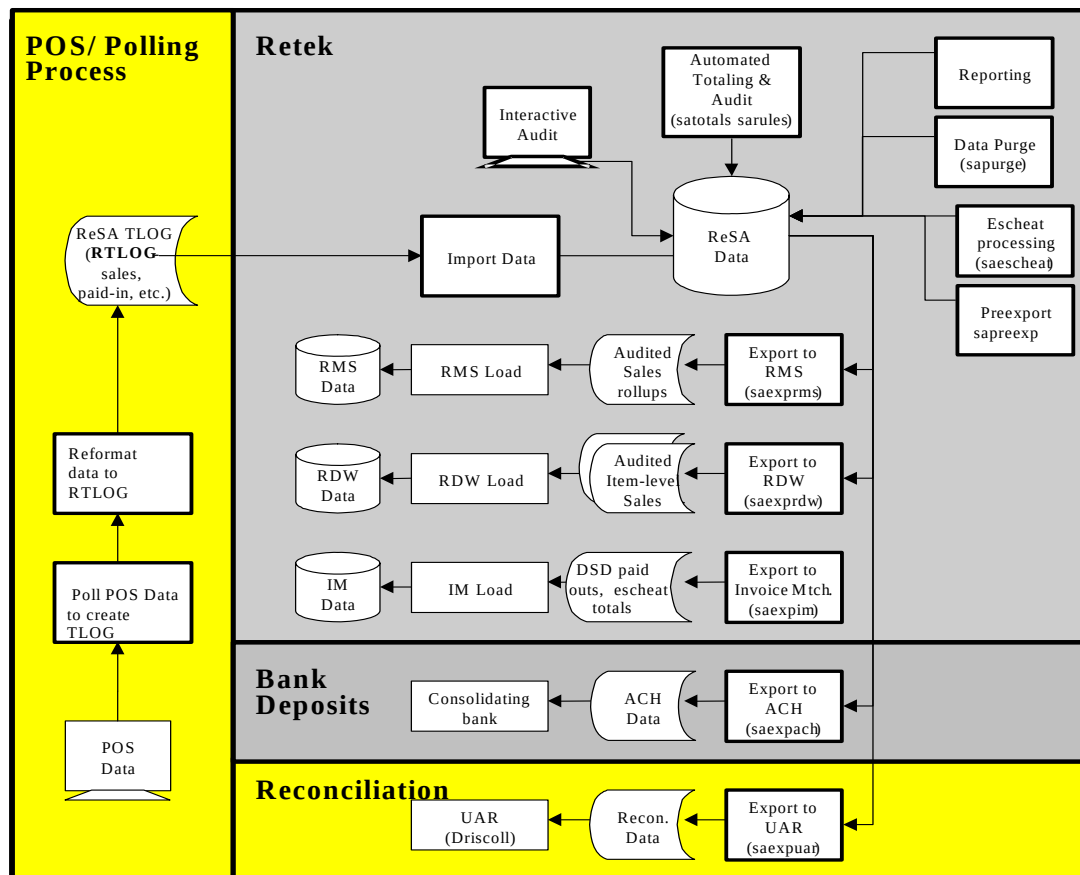
## Adjustments Received from an Application

When a retailer modifies or revises a transaction through the Sales Audit user application, numerous totals are affected through re-totalling. Whenever an application, such as UAR, returns an adjustment to a total previously received from ReSA, a package is called from either the applicable form itself or the batch module SAIMPADJ.PC. This module is responsible for updating ReSA with the change.

Upon receiving the adjustment from the application, the module identifies the total in the store day record that was exported. A revision of this total is created with the revised data. Totaling and auditing are run to recalculate store day totals. New records are created for export batch modules that send adjusted data to applications, except for the one that provided the adjustment. SAIMPADJ.PC runs after the ReSA transaction import process and before the totaling process.

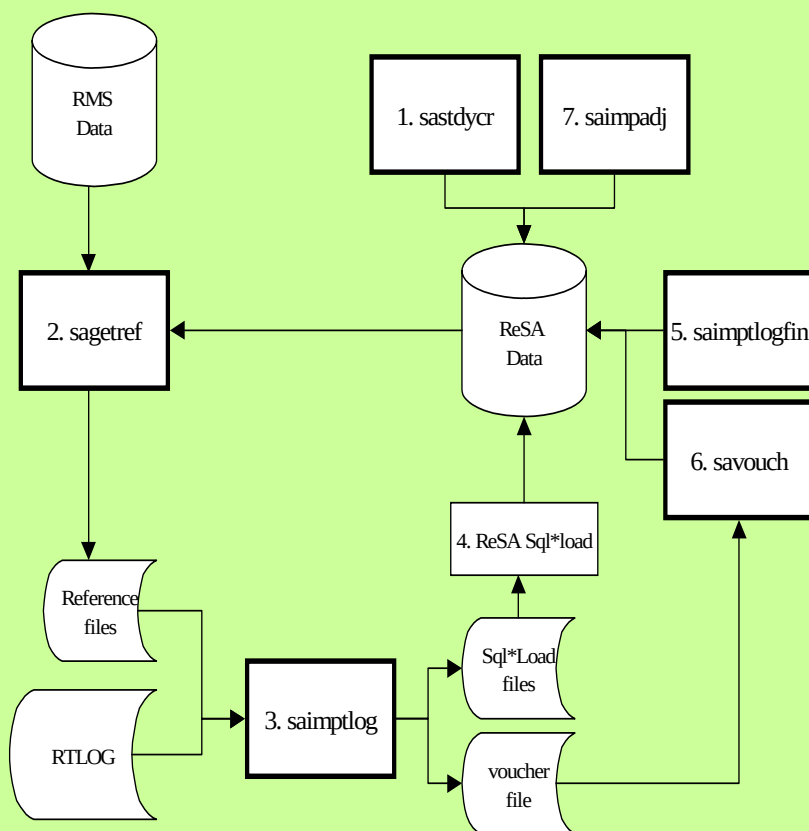
## Oracle Retail Sales Audit Dataflow Diagrams

The following two diagrams illustrate how data flows within ReSA and between ReSA and other applications. “Oracle Retail Sales Audit process dataflow” shows the entire process, and “Oracle Retail Sales Audit import process dataflow” expands the description of the “Import Data” block of the first diagram.



Oracle Retail Sales Audit process dataflow

## Import Data



Oracle Retail Sales Audit import process dataflow

## Batch Design Summary of ReSA Modules

The following list summarizes the ReSA batch modules that are involved with processing POS transaction data, audit totals and rules, exports to other applications, and modifications and adjustments.

- SAESCHEAT.PC (Sales Audit Escheated Vouchers)
- SAEXPACH.PC (Sales Audit ACH Download)
- SAEXPGL.PC (Sales Audit Export to GL)
- SAEXPIM.PC (Sales Audit Export to Invoice Matching)
- SAEXPRDW.PC (Sales Audit Export to RDW)



- SAEXPRMS.PC (Sales Audit Export to RMS)
- SAEXPUAR.PC (Universal Account Reconciliation System Export)
- SAGETREF.PC (Sales Audit Get Reference)
- SAIMPADJ.PC (Sales Audit Import Adjustments)
- SAIMPTLOG.PC (Sales Audit Import)
- SAIMPTLOGFIN.PC (Sales Audit Import)
- SAPREEXP.PC (Sales Audit Pre-Export)
- SAPURGE.PC (Sales Audit Purge)
- SARULES.PC (Sales Audit Rules)
- SASTDYCR.PC (Sales Audit Store Day Create)
- SATOTALS.PC (Sales Audit Totals)
- SAVOUCH.PC (Sales Audit Voucher Upload)

## saescheat (Sales Audit Escheated Vouchers)

### Functional Area

Oracle Retail Sales Audit (ReSA)

### Module Affected

SAESCHEAT.PC

### Design Overview

The laws of individual states and countries require a retailer to return monies for aged, unclaimed gift certificates and vouchers. This process is called 'escheatment'. SAESCHEAT.PC writes records for this data to tables that are read into ReIM by SAEXPIM.PC. The data can then be sent as invoices approved for payment to a financial application.

The SAESCHEAT batch program sets the status of vouchers that met certain State's escheats rules or expired to the proper status and produces a total for later export to Invoice Matching. This program calls the function nextEscheatSeqNo () in saescheat\_nextesn batch program, which select a block of available sequence numbers.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Sales Audit – run monthly                                                                                                                                                  |
| Scheduling Considerations | Should run after ReSA Totalling and Auding process (satotals.pc and sarules.pc) and before the export to Invoice Matching (saexpim.pc) and Sales Audit purge (sapurge.pc). |
| Pre-Processing            | N/A                                                                                                                                                                        |
| Post-Processing           | N/A                                                                                                                                                                        |

| Schedule Information | Description |
|----------------------|-------------|
| Threading Scheme     | N/A         |

**Restart/Recovery**

The logical unit of work is a store/day. The program commits when the number of store/day records processed has reached the commit\_max\_ctr.

**Locking Strategy**

This program calls the get\_lock function which uses Oracle's DMBS\_LOCK functions to ensure exclusivity and avoid lock conflicts when locking a store/day for processing.

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| SA_STORE_DAY       | Yes    | No     | No     | No     |
| SA_VOUCHER         | Yes    | No     | Yes    | No     |
| STORE              | Yes    | No     | No     | No     |
| ADDR               | Yes    | No     | No     | No     |
| SA_VOUCHER_OPTIONS | Yes    | No     | No     | No     |
| SA_ESCHEAT_VOUCHER | No     | Yes    | No     | No     |
| SA_ESCHEAT_TOTAL   | No     | Yes    | No     | No     |
| SA_ESCHEAT_OPTIONS | Yes    | No     | No     | No     |
| COMPHEAD           | Yes    | No     | No     | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**saexpach (Sales Audit ACH Download)****Functional Area**

Sales Audit

**Module Affected**

SAEXPACH.PC

## Design Overview

This module posts store/day deposit totals to the SA\_STORE\_ACH table and banks deposit totals for a given day in a standard BAI (Bank Administration Institute) format file for export to ACH (Account Clearing House). The ACH export deviates from the typical Sales Audit export in that store/days must be exported even though errors may have occurred for a given day or store (depending on the unit of work defined) and also the store/day does not need to be closed for the export to occur. The nature of the ACH process is such that as much money as possible must be sent as soon as possible to the consolidating bank. Any adjustments to the amount sent can be made via the sabnkach form.

Also, Oracle Retail assumes that there is only one total to be exported for ACH per store/day.

Deposits for store/days that are not 'F'ully loaded are not transferred to the consolidating bank. After they are fully loaded, their deposits are picked up by the next run of the program.

## Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                      |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Anytime – Sales Audit is a 24/7 system.                                                                                                                                          |
| Scheduling Considerations | This module should be run towards the end of the Sales Auditing cycle where the total (SATOTALS.PC) and rule (SARULES.PC) data are ready to be exported to the external systems. |
| Pre-Processing            | SAPREEXP.PC (preprocessing of sales auditing export modules that require totals)                                                                                                 |
| Post-Processing           | N/A                                                                                                                                                                              |
| Threading Scheme          | N/A                                                                                                                                                                              |

## Restart/Recovery

This module is in two distinct parts, with two different logical units of work. Thus, restart/recovery has to be implemented so that the first part does not get reprocessed in case the program is being restarted. Details on the implementation follow.

The first driving cursor in this module retrieves a store/day to generate ACH totals. Once the first cursor is complete, the second retrieves bank locations by account numbers.

The first Logical Unit of Work (LUW) is defined as a unique store/day combination. Records are fetched, using the first driving cursor, in batches of commit\_max\_ctr, but processed and committed one store/day at a time.

The first driving cursor fetches all store/days that are 'F'ully Loaded, whose audit status is 'A'udited, 'H'Q Errors Pending or 'S'tore Errors Pending and that are ready to be exported to ACH. Before processing starts, a write lock is obtained using get\_lock (). This driving cursor only fetches store/days with a sa\_export\_log.status of SAES\_R. After a store/day is processed, sa\_export\_log.status is set to SAES\_P so that this store/day is not selected again if the program is restarted. The commit is performed using retek\_force\_commit after each store/day was processed and sa\_export\_log updated, so as to release the lock.

In case a store/day could not be processed due to locking, then the store/day information is placed on a list (called locked store/day list) and the next store/day is processed. This list is kept in memory and is available only during processing. If the store for a store/day obtained from the first driving cursor, is on the locked store/day list, then this store/day cannot be processed. This is the case because there is a data dependency such that data from a particular store/day is dependent on data for the same store but at an earlier date. Thus, if a store/day cannot be processed, then subsequent store/days for the same store cannot be processed either. After the driving cursor returns no more data, the program attempts to process each store/day on the list two more times. If the store/day is still locked, then it is skipped entirely and a message is printed to the error log.

The second LUW is a bank account number. Again, records are fetched in batches of `commit_max_ctr`. The second driving cursor cannot retrieve information by the LUW because it is possible for the store's currency to be different from the local bank's currency. In that case, a currency conversion is needed.

For each store/day, the query should retrieve the required ACH transfer. The latter is determined by adding the estimated deposit for the next day, the adjustment to the estimate for the current day and any manual adjustment to the estimate.

Since a store can be associated with different accounts at different banks, only accounts that are consolidated should be retrieved. Since it is possible for the local bank to be in a different country than the consolidating bank, the currency of the partner should also be fetched.

Since processing is dependent on the type of account at the RDFI, the account type should be fetched by this cursor.

Due to differences in transaction processing in cases when the bank is outside the US, the partner's country should also be fetched. The results of the query should be sorted by partner country. The results of the query should also be ordered by accounts.

## **Locking Strategy**

This program is attempted to obtain a read lock on the store/day with a call to `get_lock` function. If this fails, go on to the next store/day and log the problem to the error log.

## **Security Considerations**

The fact that this program automates the transfer of funds on behalf of the user makes it a likely target for electronic theft. It must be made clear that the responsibility of electronic protection lies with the users themselves.

### **Following are some tips and recommendation to users:**

A specific user should be used to run the program. This user would be the only one (or one of a few) who has access to this program.

The umask for this user should be setup so as to prevent other users to read/write its files. This would ensure that when the output file is created, it would not be accessible to other users.

The appropriate permissions should be setup on the directory, which holds the ACH files. The most restrictive decision would be to not allow any other user to view the contents of the directory.

The password to this user should be kept confidential.

A secure means of communication should be implemented for transferring the file from where it is created to the ACH network. This may be done via encryption, or by copying the file to a disk and trusting the courier to deliver the files intact.

The ACH network needs to be secure.

### Performance Considerations

N/A

### Key Tables Affected

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| SA_ACH_INFO          | Yes    | No     | No     | No     |
| COMPHEAD             | Yes    | No     | No     | No     |
| SA_STORE_DAY         | Yes    | No     | No     | No     |
| COMPANY_CLOSED       | Yes    | No     | No     | No     |
| COMPANY_CLOSED_EXCEP | Yes    | No     | No     | No     |
| LOCATION_CLOSED      | Yes    | No     | No     | No     |
| SA_STORE_ACH         | Yes    | Yes    | Yes    | No     |
| SA_BANK_STORE        | Yes    | No     | No     | No     |
| SA_EXPORT_LOG        | Yes    | No     | Yes    | No     |
| STORE                | Yes    | No     | No     | No     |
| PARTNER              | Yes    | No     | No     | No     |
| CURRENCY_RATES       | Yes    | No     | No     | No     |
| SA_BANK_ACH          | Yes    | Yes    | Yes    | No     |

### I/O Specification

#### Output File Layout

| Record Name     | Field Name        | Field Type | Default Value | Description                                              |
|-----------------|-------------------|------------|---------------|----------------------------------------------------------|
| ACH File Header | Console Route No  | Number(9)  |               | The routing number of the consolidating bank .           |
|                 | Sender ID         | Char(10)   |               | ID used by the originator to identify itself             |
|                 | Current Date      | Char(6)    |               | Vdate<br>in 'YYMMDD' format                              |
|                 | Day Time          | Char(4)    |               | Time of file creation in HH24MM format                   |
|                 | Console Bank Name | Char(23)   |               | Name of the Originating Financial Depository Institution |

| Record Name          | Field Name        | Field Type | Default Value | Description                                                              |
|----------------------|-------------------|------------|---------------|--------------------------------------------------------------------------|
|                      | Company Name      | Char(23)   |               | The Name of the company name                                             |
|                      | Ref Code          | Char (8)   |               | Reference Code                                                           |
| ACH CCD Batch Header | Company Name      | Char(16)   |               | The Name of the company                                                  |
|                      | Comp Disc Data    | Char(20)   | NULL          | Any kind of data specific to the company                                 |
|                      | Comp Id           | Char(10)   |               | Alphanumeric code to identify the company                                |
|                      | Comp Entry Desc   | Char(10)   | “CONSO L “    | A short description from the Originator about the purpose of the entry.  |
|                      | Tomorrow          | Char(6)    |               | Vdate+1 in ‘YYMMDD’ format                                               |
|                      | Settle Date       | Char(3)    | NULL          | This is inserted by receiving ACH Operator                               |
|                      | Odfi Id           | Number(8)  |               | 8-digit routing number of the ODFI.                                      |
|                      | Batch No          | Number(7)  |               | Batch Number                                                             |
| ACH CBR Batch Header | Company Name      | Char(16)   |               | The Name of the company                                                  |
|                      | Exch Rate         | Number(15) |               | Exchange rate for the specified currency                                 |
|                      | Comp Id           | Char(10)   |               | Alphanumeric code to identify the company                                |
|                      | Comp Entry Desc   | Char(10)   | “CONSO L“     | A short description from the Originator about the purpose of the entry.  |
|                      | Partner Curr Code | Char(3)    |               | Code identifying the currency the partner uses for business transactions |
|                      | Tomorrow          | Char(6)    |               | Vdate+1 in ‘YYMMDD’ format                                               |
|                      | Settle Date       | Char(3)    | NULL          | This is inserted by receiving ACH Operator                               |
|                      | Odfi Id           | Number(8)  |               | 8-digit routing number of the ODFI.                                      |
|                      | Batch No          | Number(7)  |               | Batch Number                                                             |

| Record Name   | Field Name | Field Type | Default Value | Description                                                                                                                                               |
|---------------|------------|------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| ACH CCD Entry | Trans Code | Char(2)    |               | Code used to identify the type of debit and credit.<br>Value accepted '27','37'                                                                           |
|               | Routing No | Number(9)  |               | routing number for the bank account                                                                                                                       |
|               | Acct No    | Char(17)   |               | Account Number of the bank                                                                                                                                |
|               | Deposit    | Number(10) |               | The amount involved in the transaction* 10000 (4 implied decimal places)                                                                                  |
|               | Id         | Char(15)   | Null          | Identification Number .Optional field containing a number used by Originator to insert its own number for tracing purposes                                |
|               | Store Name | Char(22)   |               | Name of the local store                                                                                                                                   |
|               | Disc Data  | Char(2)    | Null          | Discretionary Data .Any kind of data specific to the transaction                                                                                          |
|               | Trace No   | Number(15) |               | Used to uniquely identify each entry within a batch. The first 8 digits contain the routing number of the ODFI and the other 7 contains a sequence number |
| ACH CBR Entry | Trans Code | Char(2)    |               | Code used to identify the type of debit and credit.<br>Value accepted '27','37'                                                                           |
|               | Routing No | Number(9)  |               | Routing number for the bank account                                                                                                                       |
|               | Acct No    | Char(17)   |               | Account Number of the bank                                                                                                                                |
|               | Deposit    | Number(10) |               | The amount involved in the transaction* 10000 (4 implied decimal places)                                                                                  |
|               | Id         | Char(15)   | Null          | Identification Number. Optional field containing a number used by Originator to insert its own number for tracing purposes                                |
|               | Store Name | Char(22)   |               | Name of the local store                                                                                                                                   |

| Record Name       | Field Name         | Field Type | Default Value | Description                                                                                                                                                        |
|-------------------|--------------------|------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                   | Disc Data          | Char(2)    | Null          | Discretionary Data .Any kind of data specific to the transaction                                                                                                   |
|                   | Trace No           | Number(15) |               | Used to uniquely identify each entry within a batch. The first 8 digits contain the routing number of the ODFI and the other 7 contains a sequence number          |
| ACH CBR Addendum  | Payment Info       | Char(80)   | Null          | Payment related info                                                                                                                                               |
|                   | Trace Seq No       | Number(7)  |               | This is the sequence number part of the Trace Number of the entry record to which this addendum is referring.                                                      |
| ACH Batch Control | Batch Line Count   | Number(6)  |               | The number of entries and addenda in the batch.                                                                                                                    |
|                   | Hash Count         | Number(10) |               | This is the sum of the RDFI IDs in the detail records                                                                                                              |
|                   | Total Batch Debit  | Number(12) |               | These fields contain the accumulated debit and debit for the file * 10000 (4 implied decimal places)                                                               |
|                   | Total Batch Credit | Number(12) |               | These fields contain the accumulated credit and credit for the file * 10000 (4 implied decimal places)                                                             |
|                   | Comp Id            | Char(10)   |               | An alphanumeric code identifying the company                                                                                                                       |
|                   | Auth               | Char(19)   | Null          | Message Authentication Code. The first 8 characters represent a code from the DES (Data Encryption Standard) algorithm. The remaining eleven characters are blanks |
|                   | Reserved           | Char(6)    | Null          | Reserved                                                                                                                                                           |
|                   | ODFI Id            | Number(8)  |               | 8-digit routing number of the ODFI.                                                                                                                                |
|                   | Batch No           | Number(7)  |               | Batch Number                                                                                                                                                       |
| ACH File Control  | Batch count        | Number(6)  |               | The number of batches sent in the file.                                                                                                                            |



| Record Name         | Field Name         | Field Type | Default Value | Description                                                                                                                                                                               |
|---------------------|--------------------|------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                     | Block count        | Number(6)  |               | The number of physical blocks in the file, including both File Header and File Control Records. This is the ceiling of the number of records divided by the blocking factor, which is 10. |
|                     | Entry count        | Number(8)  |               | The number of entries and addenda in the file.                                                                                                                                            |
|                     | Total hash count   | Number(10) |               | This is the sum of the Entry Hash fields on the Batch Control Records.                                                                                                                    |
|                     | Total file debit   | Number(12) |               | These fields contain the accumulated debit and debit for the file * 10000 (4 implied decimal places)                                                                                      |
|                     | Total file credit, | Number(12) |               | These fields contain the accumulated credit and credit for the file * 10000 (4 implied decimal places)                                                                                    |
|                     | Reserved           | Char(39)   | Null          | Reserved                                                                                                                                                                                  |
| ACH Completed Block | End string         | Char(94)   |               | Mark the end of the file: a string of 94 '9's.                                                                                                                                            |

## saexpgl (Sales Audit Export to GL)

### Functional Area

Oracle Retail Sales Audit (ReSA)

### Module Affected

SAEXPGL.PC

### Design Overview

The purpose of SAEXPGL batch module is to post all properly configured user defined ReSA totals to the User defined General ledger application (Oracle or PeopleSoft). Totals without errors are posted to the appropriate accounting ledger, as defined in the Sales Audit Oracle cross-reference user module. Depending on the unit of work system option, the data is sent at either the store/day or individual total level. Newly revised totals that are already posted to the ledger have their previous revision reversed, and the new total posted to the appropriate accounts. Transactions that are from previous periods are posted to the current period.

This version of the program is meant for the interface between RMS 12.0 and Oracle Financials.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                 |
|---------------------------|-------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Anytime. The processing cycle of the Sales Audit is a 24/7 system.                                          |
| Scheduling Considerations | This program should run after the ReSA Totaling process (satotals.pc) and Audit Rules process (sarules.pc). |
| Pre-Processing            | N/A                                                                                                         |
| Post-Processing           | N/A                                                                                                         |
| Threading Scheme          | N/A                                                                                                         |

### Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records are fetched, updated and inserted in batches the size of commit max counter. Only one commit is done after a store/day is completely processed. A call to release\_lock() performs a commit.

### Locking Strategy

This program obtains a read lock on the store/day with a call to get\_lock. Failure to obtain a lock results in a non-fatal error with the program continuing on to the next store/day. An entry is written to the error log for the store/day that could not be locked.

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table               | Select | Insert | Update | Delete |
|---------------------|--------|--------|--------|--------|
| FIF_GL_SETUP        | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS      | Yes    | No     | No     | No     |
| SA_EXPORT_LOG       | Yes    | No     | Yes    | No     |
| STORE               | Yes    | No     | No     | No     |
| SA_FIF_GL_CROSS_REF | Yes    | No     | No     | No     |
| STG_FIF_GL_DATA     | No     | Yes    | No     | No     |

| Table       | Select | Insert | Update | Delete |
|-------------|--------|--------|--------|--------|
| IF_ERRORS   | No     | Yes    | No     | No     |
| SA_EXPORTED | No     | Yes    | Yes    | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**saexpim (Sales Audit Export to Invoice Matching)****Functional Area**

Oracle Retail Sales Audit (ReSA)

**Module Affected**

SAEXPIM.PC

**Design Overview**

The purpose of this Invoice Matching (SAEXPIM) program is to support invoices from Direct Store Delivery and Escheatment sales audit transactions. Direct Store Delivery invoices refer to products or services that are delivered to the store and paid out at the store. This program takes DSD invoices that were staged to the SA\_TRAN\_HEAD table by the SAIMPTLOG.PC program and moves them into the INVC\_HEAD table. All DSD transactions are assumed paid. They can be assumed received if there is a proof of delivery number listed on them. Transactions with a vendor invoice ID or a proof of delivery number should be matched to any existing invoice in INVC\_HEAD, and that invoice updated with the new information being interfaced. Invoices that do not match an existing invoice in INVC\_HEAD need to be inserted. Each transaction is exported to INVC\_HEAD table only once.

The Sales Audit Transaction type used to identify invoices for Direct Store Delivery transactions are "Paid Out". Transaction types are stored on the codes tables with a code\_type = 'TRAT'. The Paid Out transaction has a code of 'PAIDOU'. The Sales Audit sub-transaction types are used to identify whether the invoice is an "Expense Vendor Payout" or a "Merchandise Vendor Payout". These types are stored on the codes table with a code\_type = 'TRAS'. The codes are 'EV' for Expense Vendor Payout and 'MV' for Merchandise Vendor Payout. Any Paid Out transaction with a sub transaction type of Expense Vendor creates a non-merchandise invoice and causes a record to be written to the INVC\_NON\_MERCH table. ReSA stores non-merchandise codes in the reason\_code field on sa\_tran\_head. Valid values for these reason codes should correspond to the codes stored on the non\_merch\_code\_head table.

In addition to DSD invoices, this program also interfaces Escheatment totals to Invoice Matching. Escheatment is the process where an unredeemed gift certificate/voucher or credit voucher is, after a set period of time, paid out as income to the issuing Retailer or in some states, the State receives this escheatment income. ReSA is the governing system that determines who receives this income, but Invoice Matching sends the totals, with the related Partner, to Accounts Payable. Escheatment information is stored on the ReSA SA\_TOTALS table and is used to create non-merchandise invoices in Invoice Matching. These invoices are assumed not paid.

### Scheduling Constraints

| Schedule Information      | Description                                                                             |
|---------------------------|-----------------------------------------------------------------------------------------|
| Processing Cycle          | Daily – Anytime, Sales Audit is a 24/7 system.                                          |
| Scheduling Considerations | This module should be executed after the ReSA transaction import process after sapreexp |
| Pre-Processing            | N/A                                                                                     |
| Post-Processing           | N/A                                                                                     |
| Threading Scheme          | N/A                                                                                     |

### Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records are fetched, updated and inserted based on the commit\_max\_ctr specified on the RESTART\_CONTROL table. Only two commits are done, one to establish the store/day lock and another at the end, to release the lock after a store/day is completely processed. In case of failure, all work done is rolled back to the point right after the call to get\_lock and releases the lock. Thus, the rollback segment should be large enough to hold all inserts into sa\_exported for one store\_day.

### Locking Strategy

This program is attempted to obtain a read lock on the store/day with a call to get\_lock. If this fails, go on to the next store/day and log the problem to the error log.

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |
| SA_STORE_DAY   | Yes    | No     | No     | No     |
| SA_EXPORT_LOG  | Yes    | No     | No     | No     |

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| SA_TRAN_HEAD    | Yes    | No     | No     | No     |
| SA_TRAN_TENDER  | Yes    | No     | No     | No     |
| SA_EXPORTED     | Yes    | Yes    | No     | No     |
| INVC_HEAD       | Yes    | Yes    | Yes    | No     |
| INVC_NON_MERCH  | No     | Yes    | Yes    | No     |
| INVC_XREF       | No     | Yes    | No     | No     |
| TERMS           | Yes    | No     | No     | No     |
| SUPS            | Yes    | No     | No     | No     |
| PARTNER         | Yes    | No     | No     | No     |
| CURRENCY_RATES  | Yes    | No     | No     | No     |
| ADDR            | Yes    | No     | No     | No     |
| SA_ERROR        | Yes    | No     | No     | No     |
| SA_ERROR_IMPACT | Yes    | No     | No     | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**saexprdw (Sales Audit Export to RDW)****Functional Area**

Oracle Retail Sales Audit (ReSA)

**Module Affected**

SAEXPRDW.PC

**Design Overview**

The purpose of this batch module is to fetch all corrected sale and return transactions that do not have RDW errors from the Oracle Retail Sales Audit (ReSA) database tables for transmission to the Oracle Retail Merchandising System Data Warehouse (RDW). The data is sent at the store day level. If the transaction has a status of Deleted and if it is previously Transmitted, a reversal of the transaction is sent. Four files of type RDWT, RDWF, RDWS and RDWC is created for each store\_day.

**Scheduling Constraints**

| Schedule Information | Description                     |
|----------------------|---------------------------------|
| Processing Cycle     | Anytime – ReSA is a 24/7 system |

| Schedule Information      | Description                                                                               |
|---------------------------|-------------------------------------------------------------------------------------------|
| Scheduling Considerations | This runs after auditors have made corrections to the data.                               |
| Pre-Processing            | sagetref.pc to get waste data, and saimptlog.pc and saimptlogfin.pc to get post-void data |
| Post-Processing           | stsldmat.pc, tldmat.pc, lptotclat.pc, lptotldat.pc                                        |
| Threading Scheme          | Multi-threaded by store                                                                   |

### Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records are fetched, updated and inserted based on the commit\_max\_ctr. Only two commits are done: one to establish the store/day lock and another at the end, to release the lock after a store/day is completely processed. The RDWT, RDWF, RDWS and RDWC formatted output files are created with temporary names and renamed just before the end of store/day commit.

In case of a failure, all the work done is rolled back to the point right after the call to get\_lock() and releases the lock. Thus, the rollback segment should be large enough to hold all inserts into sa\_exported for one store/day.

### Locking Strategy

This program attempts to obtain a read lock on the store/day with a call to get\_lock(). If this fails, go on to the next store/day and log the problem to the error log.

### Security Considerations

Credit card numbers and other customer information are present in the output files. Access to these files is controlled only by the Unix permissions that these files have.

### Performance Considerations

N/A

### Key Tables Affected

| Table                    | Select | Insert | Update | Delete |
|--------------------------|--------|--------|--------|--------|
| SA_STORE_DAY             | Yes    | No     | No     | No     |
| SA_EXPORT_LOG            | Yes    | No     | No     | No     |
| V_RESTART_STORE          | Yes    | No     | No     | No     |
| SA_STORE_PRICE_HIST_TEMP | No     | Yes    | No     | No     |
| SA_TRAN_HEAD             | Yes    | No     | No     | No     |
| SA_CUSTOMER              | Yes    | No     | No     | No     |
| SA_STORE_EMP             | Yes    | No     | No     | No     |
| SA_ERROR                 | Yes    | No     | No     | No     |
| SA_ERROR_IMPACT          | Yes    | No     | No     | No     |

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| SA_EXPORTED            | Yes    | No     | No     | No     |
| SA_TRAN_HEAD_REV       | Yes    | No     | No     | No     |
| SA_EXPORTED_REV        | Yes    | No     | No     | No     |
| SA_TRAN_ITEM           | Yes    | No     | No     | No     |
| SA_TRAN_ITEM_REV       | Yes    | No     | No     | No     |
| SA_TRAN_DISC           | Yes    | No     | No     | No     |
| SA_TRAN_DISC_REV       | Yes    | No     | No     | No     |
| SA_TRAN_TENDER         | Yes    | No     | No     | No     |
| SA_VOUCHER             | Yes    | No     | No     | No     |
| SA_TRAN_TENDER_REV     | Yes    | No     | No     | No     |
| SA_STORE_DAY_READ_LOCK | No     | Yes    | No     | Yes    |

## I/O Specification

### Output File Layout

Four output files are created for each store\_day.

- RDWT - Transaction File
- RDWF - Form of Payment (Tender) file
- RDWS - Store Totals output file
- RDWC - Cashier output File

Each output file is converted into a format for loading into RDW by the resa2rdw Perl script.

## Oracle Retail Sales Audit (ReSA) – File Layout – RDW

File layouts for interface between sales audit and data warehouse.

Char fields are left justified and blank filled.

Number fields are right justified and zero filled. They can contain only numbers.

Numeric fields are left justified and blank filled. They can contain only numbers.

### Transaction Item Information produced by saexprdw.pc

| Record Name | Field Name                  | Field Type | Default Value                | Description                                       | Required |
|-------------|-----------------------------|------------|------------------------------|---------------------------------------------------|----------|
| File Header | File Type Record Descriptor | Char(5)    | FHEAD                        | Identifies file record type                       |          |
|             | File Line Identifier        | Number(10) | specified by external system | ID of current line being processed by input file. | Yes      |

| Record Name        | Field Name                  | Field Type | Default Value                | Description                                                                                                                                                                                              | Required         |
|--------------------|-----------------------------|------------|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
|                    | File Type Definition        | Char(4)    | RDWT                         | Identifies file as 'RDW Transaction file'                                                                                                                                                                | Yes              |
|                    | File Create Date            | Number(14) | create date                  | Date file was written by external system. Format YYYYMMDDHH24MISS                                                                                                                                        | Yes              |
| Transaction Header | File Type Record Descriptor | Char(5)    | THEAD                        | Identifies transaction record type                                                                                                                                                                       |                  |
|                    | File Line Identifier        | Number(10) | specified by external system | ID of current line being processed by input file.                                                                                                                                                        | Yes              |
|                    | Business date               | Number(8)  |                              | Format YYYYMMDD (Note, This is the date the RDW considers the transaction date)                                                                                                                          | Yes              |
|                    | Transaction Date            | Number(14) | transaction date             | Date sale/return transaction was processed at the POS. Format YYYYMMDDHH24MISS (Note, the RDW only uses the HH24MI part of this date)                                                                    | Yes              |
|                    | Location                    | Number(10) | specified by external system | Store or warehouse identifier                                                                                                                                                                            | Yes              |
|                    | Register ID                 | Char(5)    |                              | The register identifier                                                                                                                                                                                  | Yes, -1 for null |
|                    | Banner ID                   | Char(4)    |                              | The unique identifier of the banner.                                                                                                                                                                     | Yes, -1 for null |
|                    | Line Media ID               | Char(10)   |                              | The identifier of the media for the order line. For non-merchandise items, such as Shipping & Handling, Service Lines, and gift certificates, the media code are that of the order line it's associated. | Yes, -1 for null |
|                    | Selling Item ID             | Char(25)   |                              | The unique identifier of a selling item.                                                                                                                                                                 | Yes, -1 for null |
|                    | Customer Order Header ID    | Char(30)   |                              | The unique identifier of a customer order.                                                                                                                                                               | Yes, -1 for null |



| Record Name | Field Name                  | Field Type  | Default Value | Description                                                                                                                                                | Required                               |
|-------------|-----------------------------|-------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------|
|             | Customer Order Line ID      | Char(30)    |               | The identifier of a customer order line. For a Value Added Service, like monogramming, this is the line number for the item which the service was applied. | Yes, -1 for null                       |
|             | Customer Order Create Date  | Number(8)   |               | The customer order creation date                                                                                                                           | Yes, 'transaction date' for null       |
|             | Cashier Identifier          | Char(10)    |               | The cashier number. This is the unique employee number.                                                                                                    | Yes, -1 for null                       |
|             | Salesperson Identifier      | Char(10)    |               | The salesperson number. This is the unique employee number.                                                                                                | Yes, -1 for null                       |
|             | Customer ID Type            | Char(6)     |               | The type of ID number used by this customer.                                                                                                               | Yes, -1 for null                       |
|             | Customer ID Number          | Char(16)    |               | Customer id associated with the transaction.                                                                                                               | Yes, -1 for null                       |
|             | Transaction Number          | Number(10)  |               | The unique transaction reference number generated by the POS.                                                                                              | Yes                                    |
|             | Original Register ID        | Char(5)     |               | Register ID of the original transaction.                                                                                                                   | Yes for a transaction type of 'PVOID'. |
|             | Original Transaction Number | Number(10)  |               | Transaction number of the original transaction.                                                                                                            | Yes for a transaction type of 'PVOID'. |
|             | Transaction Header Number   | Numeric(20) |               | Unique reference used within sales audit to represent the date/store/register/tran_no                                                                      | Yes                                    |
|             | Revision number             | Number(3)   |               | Number used to identify the version of the transaction being sent.                                                                                         | Yes                                    |

| Record Name | Field Name                | Field Type  | Default Value                            | Description                                                                                                                          | Required                               |
|-------------|---------------------------|-------------|------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------|
|             | Sales Sign                | Char(1)     | 'P' - positive<br>'N' – negative         | Determines if the Total Sales Quantity and Total Sales Value are positive or negative.                                               | Yes                                    |
|             | Transaction Type          | Char(6)     |                                          | Transaction type code                                                                                                                | Yes                                    |
|             | Sub Transaction Type      | Char(6)     |                                          | The Sub Transaction type                                                                                                             | Yes, -1 for null                       |
|             | Retail Type               | Char(1)     | 'R'egular,<br>'P'romo, or<br>'C'learance |                                                                                                                                      | Yes                                    |
|             | Item_Seq_No               | Number(4)   |                                          | The order in which items were entered during the transaction.                                                                        | No                                     |
|             | Employee Number (Cashier) | Char(10)    |                                          | Employee identification number. This is only populated if the sub transaction type is 'EMP'.                                         | Yes, -1 for null                       |
|             | Receipt Indicator         | Char(1)     |                                          | Flag that identifies returns that are processed without a receipt. This field is only populated if the transaction type is 'RETURN'. | No                                     |
|             | Reason Code               | Char(6)     |                                          | A reason is required with a Paid In/Out transaction type, and optional with a return transaction.                                    | Yes, -1 for null                       |
|             | Vendor number             | Numeric(10) |                                          | This is only populated when the paid in code is Expense Vendor                                                                       | No                                     |
|             | Item Type                 | Char(6)     | item type identifier                     | Type of item sold, 'ITEM', 'REF', 'GCN' (gift certificate number), or 'NMITEM'                                                       | No                                     |
|             | Item                      | Char(25)    |                                          | ID number of the item or gift certificate.                                                                                           | No. Required if Item Type is not null. |

| Record Name | Field Name              | Field Type | Default Value                    | Description                                                                                                                                                             | Required                                                                                        |
|-------------|-------------------------|------------|----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
|             | Ref Item                | Char(25)   |                                  | Sub-transaction level item                                                                                                                                              | No. Also, this field can never be populated without a transaction level item in the item field. |
|             | Taxable Indicator       | Char(1)    |                                  | Taxable/non-taxable status indicator                                                                                                                                    | No                                                                                              |
|             | Entry/mode              | Char(6)    |                                  | Indicator that identifies whether the item was scanned or manually entered                                                                                              | No                                                                                              |
|             | Department              | Number(4)  |                                  | Department of item sold or returned. Yes need to validate if using ReSA.                                                                                                | No                                                                                              |
|             | Class                   | Number(4)  |                                  | Class of item sold or returned. Yes need to validate if using ReSA.                                                                                                     | No                                                                                              |
|             | Subclass                | Number(4)  |                                  | Subclass of item sold or returned. Yes need to validate if using ReSA.                                                                                                  | No                                                                                              |
|             | Total Sales Quantity    | Number(12) |                                  | Number of units sold at a particular location with 4 implied decimal places.                                                                                            | No                                                                                              |
|             | Total Transaction Value | Number(20) |                                  | Sales value, net sales value of goods sold/returned with 4 implied decimal places.                                                                                      | No                                                                                              |
|             | Override Reason         | Char(6)    |                                  | This column is populated when an item's price is overridden at the POS to define why it was overridden. This is also always sent if the transaction originated in RCOM. | Yes, -1 for null                                                                                |
|             | Return Reason           | Char(6)    |                                  | The reason an item was returned.                                                                                                                                        | Yes, -1 for null                                                                                |
|             | Total original sign     | Char(1)    | 'P' - positive<br>'N' - negative |                                                                                                                                                                         | No                                                                                              |

| Record Name        | Field Name                   | Field Type  | Default Value                | Description                                                                                                                                                                                                         | Required |
|--------------------|------------------------------|-------------|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|
|                    | Total Original Sales Value   | Number(20)  |                              | This column is populated when the item's price was overridden at the POS and the item's original unit retail is known. This is always written when the transaction originated in RCOM. This has 4 implied decimals. | No       |
|                    | Weather                      | Char(6)     |                              | For transaction types of 'COND', this field stores the type of weather for the store-day.                                                                                                                           | No       |
|                    | Temperature                  | Char(6)     |                              | For transaction types of 'COND', this field stores the type of temperature for the store-day.                                                                                                                       | No       |
|                    | Traffic                      | Char(6)     |                              | For transaction types of 'COND', this field stores the type of traffic for the store-day.                                                                                                                           | No       |
|                    | Construction                 | Char(6)     |                              | For transaction types of 'COND', this field stores information regarding any construction on that store-day.                                                                                                        | No       |
|                    | Drop Shipment Indicator      | Char(1)     | 'Y' or 'N'                   | Indicates whether item is involved in a drop shipment.                                                                                                                                                              | No       |
| Transaction Detail | File Type Record Descriptor  | Char(5)     | TDETL                        | Identifies transaction record type                                                                                                                                                                                  |          |
|                    | File Line Identifier         | Number(10)  | specified by external system | ID of current line being processed by input file.                                                                                                                                                                   | Yes      |
|                    | Discount Type                | Char(6)     |                              | Code for discount type from code_detail, code_type = 'SADT'                                                                                                                                                         | No       |
|                    | Promotional Transaction Type | Char(6)     |                              | Code for promotional type from code_detail, code_type = 'PRMT'                                                                                                                                                      | Yes      |
|                    | Promotion Number             | Numeric(10) | promotion number             | Promotion number from the RMS                                                                                                                                                                                       | No       |

| Record Name         | Field Name                  | Field Type  | Default Value                    | Description                                                               | Required                              |
|---------------------|-----------------------------|-------------|----------------------------------|---------------------------------------------------------------------------|---------------------------------------|
|                     | Promotion Component Number  | Numeric(10) | Promo_comp_id from RPM           |                                                                           | Required if it is a promotional sale. |
|                     |                             |             |                                  |                                                                           |                                       |
|                     | Coupon Number               | Char(16)    |                                  |                                                                           | Yes if Discount Type is 'SCOUP'.      |
|                     | Coupon Reference Number     | Char(16)    |                                  |                                                                           | No                                    |
|                     | Sales Quantity              | Number(12)  |                                  | Number of units sold in this prom type with 4 implied decimal places.     | No                                    |
|                     | Transaction Sign            | Char(1)     | 'P' - positive<br>'N' - negative |                                                                           | Yes                                   |
|                     | Transaction Value           | Number(20)  |                                  | Value of units sold in this promotion type with 4 implied decimal places. | Yes                                   |
|                     | Discount Value              | Number(20)  |                                  | Value of discount given in this prom type with 4 implied decimal places.  | Yes                                   |
| Transaction Trailer | File Type Record Descriptor | Char(5)     | TTAIL                            | Identifies file record type                                               |                                       |
|                     | File Line Identifier        | Number(10)  | specified by external system     | ID of current line being processed by input file.                         | Yes                                   |
|                     | Transaction Count           | Number(6)   | specified by external system     | Number of TDETL records in this transaction set                           | Yes                                   |
| File Trailer        | File Type Record Descriptor | Char(5)     | FTAIL                            | Identifies file record type                                               |                                       |
|                     | File Line Identifier        | Number(10)  | specified by external system     | ID of current line being processed by input file.                         | Yes                                   |

| Record Name | Field Name          | Field Type | Default Value | Description                                                                                 | Required |
|-------------|---------------------|------------|---------------|---------------------------------------------------------------------------------------------|----------|
|             | File Record Counter | Number(10) |               | Number of records/transactions processed in current file (only records between head & tail) | Yes      |

**Transaction Item Information Produced by saexprdw.pc after  
Translation by resa2rdw**

| Record Name | Field Name       | Field Type | Default Value                | Description                                                                                                                                                                                      | Required         |
|-------------|------------------|------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
|             | Business date    | Number(8)  |                              | Format YYYYMMDD                                                                                                                                                                                  | Yes              |
|             | Transaction Date | Number(14) | transaction date             | Date sale/return transaction was processed at the POS. Format YYYYMMDDHH24MISS                                                                                                                   | Yes              |
|             | Location         | Number(10) | specified by external system | Store or warehouse identifier                                                                                                                                                                    | Yes              |
|             | Register ID      | Char(5)    |                              | The register identifier                                                                                                                                                                          | Yes, -1 for null |
|             | Banner ID        | Char(4)    |                              | The unique identifier of the banner.                                                                                                                                                             | Yes, -1 for null |
|             | Line Media ID    | Char(10)   |                              | The identifier of the order line media. For non-merchandise items, such as Shipping & Handling, Service Lines, and gift certificates, the media code are that of the order line it's associated. | Yes, -1 for null |
|             | Selling Item ID  | Char(25)   |                              | The unique identifier of a selling item.                                                                                                                                                         | Yes, -1 for null |

| Record Name | Field Name                  | Field Type | Default Value | Description                                                                                                                                                 | Required                               |
|-------------|-----------------------------|------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------|
|             | Customer Order Header ID    | Char(30)   |               | The unique identifier of a customer order.                                                                                                                  | Yes, -1 for null                       |
|             | Customer Order Line ID      | Char(30)   |               | The identifier of a customer order line. For a Value Added Service, like monogramming, this is the line number for the item, which the service was applied. | Yes, -1 for null                       |
|             | Customer Order Create Date  | Number(8)  |               | The customer order creation date                                                                                                                            | Yes, 'transaction date' for null       |
|             | Cashier Identifier          | Char(10)   |               | The cashier number. This is the unique employee number.                                                                                                     | Yes, -1 for null                       |
|             | Salesperson Identifier      | Char(10)   |               | The salesperson number. This is the unique employee number.                                                                                                 | Yes, -1 for null                       |
|             | Customer ID Type            | Char(6)    |               | The type of ID number used by this customer.                                                                                                                | Yes, -1 for null                       |
|             | Customer ID Number          | Char(16)   |               | Customer id associated with the transaction.                                                                                                                | Yes, -1 for null                       |
|             | Transaction Number          | Number(10) |               | The unique transaction reference number generated by the POS.                                                                                               | Yes                                    |
|             | Original Register ID        | Char(5)    |               | Register ID of the original transaction.                                                                                                                    | Yes for a transaction type of 'PVOID'. |
|             | Original Transaction Number | Number(10) |               | Transaction number of the original transaction.                                                                                                             | Yes for a transaction type of 'PVOID'. |

| Record Name | Field Name                | Field Type  | Default Value                            | Description                                                                                                                           | Required         |
|-------------|---------------------------|-------------|------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|------------------|
|             | Transaction Header Number | Numeric(20) |                                          | Unique reference used within sales audit to represent the date/store/register/transaction_no                                          | Yes              |
|             | Revision number           | Number(3)   |                                          | Number used to identify the version of the transaction being sent.                                                                    | Yes              |
|             | Sales Sign                | Char(1)     | 'P' - positive<br>'N' - negative         | Determines if the Total Sales Quantity and Total Sales Value are positive or negative.                                                | Yes              |
|             | Transaction Type          | Char(6)     |                                          | Transaction type code                                                                                                                 | Yes              |
|             | Sub Transaction Type      | Char(6)     |                                          | The Sub Transaction type                                                                                                              | Yes, -1 for null |
|             | Retail Type               | Char(1)     | 'R'egular,<br>'P'romo, or<br>'C'learance |                                                                                                                                       | Yes              |
|             | Item_Seq_No               | Number(4)   |                                          | The order in which items were entered during the transaction.                                                                         | No               |
|             | Employee Number (Cashier) | Char(10)    |                                          | Employee identification number. This is only populated if the sub transaction type is 'EMP'.                                          | Yes, -1 for null |
|             | Receipt Indicator         | Char(1)     |                                          | Flag that identifies returns that were processed without a receipt. This field is only populated if the transaction type is 'RETURN'. | No               |



| Record Name | Field Name        | Field Type  | Default Value        | Description                                                                                       | Required                                                                                        |
|-------------|-------------------|-------------|----------------------|---------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
|             | Reason Code       | Char(6)     |                      | A reason is required with a Paid In/Out transaction type, and optional with a return transaction. | Yes, -1 for null                                                                                |
|             | Vendor number     | Numeric(10) |                      | This is only populated when the paid in code is Expense Vendor                                    | No                                                                                              |
|             | Item Type         | Char(6)     | item type identifier | Type of item sold, 'ITEM', 'REF', 'GCN' (gift certificate number), or 'NMITEM'                    | No                                                                                              |
|             | Item              | Char(25)    |                      | ID number of the item or gift certificate.                                                        | No. Required if Item Type is not null.                                                          |
|             | Ref Item          | Char(25)    |                      | Sub-transaction level item                                                                        | No. Also, this field can never be populated without a transaction level item in the item field. |
|             | Taxable Indicator | Char(1)     |                      | Taxable/non-taxable status indicator                                                              | No                                                                                              |
|             | Entry/mode        | Char(6)     |                      | Indicator that identifies whether the item was scanned or manually entered                        | No                                                                                              |
|             | Department        | Number(4)   |                      | Department of item sold or returned. Yes need to validate if using ReSA.                          | No                                                                                              |
|             | Class             | Number(4)   |                      | Class of item sold or returned. Yes need to validate if using ReSA.                               | No                                                                                              |
|             | Subclass          | Number(4)   |                      | Subclass of item sold or returned. Yes need to validate if using ReSA.                            | No                                                                                              |

| Record Name | Field Name                 | Field Type | Default Value                   | Description                                                                                                                                                                                                    | Required         |
|-------------|----------------------------|------------|---------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
|             | Total Sales Quantity       | Number(12) |                                 | Number of units sold at a particular location with 4 implied decimal places.                                                                                                                                   | No               |
|             | Total Transaction Value    | Number(20) |                                 | Sales value, net sales value of goods sold/returned with 4 implied decimal places.                                                                                                                             | No               |
|             | Override Reason            | Char(6)    |                                 | This column is populated when an item price is overridden at the POS to define why it was overridden. This is always sent if the transaction originated in RCOM.                                               | Yes, -1 for null |
|             | Return Reason              | Char(6)    |                                 | The reason an item was returned.                                                                                                                                                                               | Yes, -1 for null |
|             | Total original sign        | Char(1)    | 'P'- positive<br>'N' – negative |                                                                                                                                                                                                                | No               |
|             | Total Original Sales Value | Number(20) |                                 | This column is populated when the item's price was overridden at the POS and the item's original unit retail is known. This is always sent if the transaction originated in RCOM. This has 4 implied decimals. | No               |
|             | Weather                    | Char(6)    |                                 | For transaction types of 'COND', this field stores the type of weather for the store-day.                                                                                                                      | No               |

| Record Name | Field Name                   | Field Type  | Default Value          | Description                                                                                                  | Required                              |
|-------------|------------------------------|-------------|------------------------|--------------------------------------------------------------------------------------------------------------|---------------------------------------|
|             | Temperature                  | Char(6)     |                        | For transaction types of 'COND', this field stores the type of temperature for the store-day.                | No                                    |
|             | Traffic                      | Char(6)     |                        | For transaction types of 'COND', this field stores the type of traffic for the store-day.                    | No                                    |
|             | Construction                 | Char(6)     |                        | For transaction types of 'COND', this field stores information regarding any construction on that store-day. | No                                    |
|             | Drop Shipment Indicator      | Char(1)     | 'Y' or 'N'             | Indicates whether item is involved in a drop shipment.                                                       | No                                    |
|             |                              |             |                        |                                                                                                              |                                       |
|             |                              |             |                        |                                                                                                              |                                       |
|             | Discount Type                | Char(6)     |                        | Code for discount type from code_detail, code_type = 'SADT'                                                  | No                                    |
|             | Promotional Transaction Type | Char(6)     |                        | Code for promotional type from code_detail, code_type = 'PRMT'                                               | Yes                                   |
|             | Promotion Number             | Numeric(10) | promotion number       | Promotion number from the RMS                                                                                | No                                    |
|             | Promotion Component Number   | Numeric(10) | Promo_comp_id from RPM |                                                                                                              | Required if it is a promotional sale. |
|             | Coupon Number                | Char(16)    |                        |                                                                                                              | Yes if Discount Type is 'SCoup'.      |
|             | Coupon Reference Number      | Char(16)    |                        |                                                                                                              | No                                    |

| Record Name | Field Name        | Field Type | Default Value                    | Description                                                               | Required |
|-------------|-------------------|------------|----------------------------------|---------------------------------------------------------------------------|----------|
|             | Sales Quantity    | Number(12) |                                  | Number of units sold in this prom type with 4 implied decimal places.     | No       |
|             | Transaction Sign  | Char(1)    | 'P' - positive<br>'N' – negative |                                                                           | Yes      |
|             | Transaction Value | Number(20) |                                  | Value of units sold in this promotion type with 4 implied decimal places. | Yes      |
|             | Discount Value    | Number(20) |                                  | Value of discount given in this prom type with 4 implied decimal places.  | Yes      |

### RDW Form of Payment File

| Record Name | Field Name                  | Field Type  | Default Value                | Description                                                              | Required |
|-------------|-----------------------------|-------------|------------------------------|--------------------------------------------------------------------------|----------|
| File Header | File Type Record Descriptor | Char(5)     | FHEAD                        | Identifies file record type                                              |          |
|             | File Line Identifier        | Number(10)  | specified by external system | ID of current line being processed by input file.                        | Yes      |
|             | File Type Definition        | Char(4)     | RDWF                         | Identifies file as 'RDW Form of Payment (Tender) file'                   | Yes      |
|             | File Create Date            | Numeric(14) | create date                  | date file was written by external system. Format<br>YYYYMMDDH<br>H24MISS | Yes      |
| File Detail | File Type Record Descriptor | Char(5)     | FDETL                        | Identifies file record type                                              |          |
|             | File Line Identifier        | Number(10)  | specified by external system | ID of current line being processed by input file.                        | Yes      |

| Record Name | Field Name                  | Field Type  | Default Value                   | Description                                                                                      | Required         |
|-------------|-----------------------------|-------------|---------------------------------|--------------------------------------------------------------------------------------------------|------------------|
|             | Business date               | Numeric(8)  |                                 | Format<br>YYYYMMDD                                                                               | Yes              |
|             | Transaction Date            | Numeric(14) | transaction date                | Date sale/return transaction was processed at the POS. Format<br>YYYYMMDDH<br>H24MISS            | Yes              |
|             | Location                    | Number(10)  | specified by external system    | Store or warehouse identifier.                                                                   | Yes              |
|             | Cashier Identifier          | Char(10)    |                                 | The cashier number. This is the unique employee number.                                          | Yes, -1 for null |
|             | Register Identifier         | Char(5)     |                                 |                                                                                                  | Yes, -1 for null |
|             | Sales Sign                  | Char(1)     | 'P'- positive<br>'N' – negative | Determines if the Total Sales Quantity and Total Sales Value are positive or negative.           | Yes              |
|             | Transaction Sequence Number | Numeric(20) |                                 | Unique reference used within sales audit to represent the date/store/register/transaction number | Yes              |
|             | Revision number             | Number(3)   |                                 | Number used to identify the version of the transaction being sent.                               | Yes              |
|             | Transaction Type            | Char(6)     |                                 | Transaction type code.                                                                           | Yes              |
|             | Tender type group           | Char(6)     |                                 |                                                                                                  | Yes              |
|             | Tender type id              | Numeric(6)  |                                 | Tender type code.                                                                                | Yes              |
|             | Tender amount               | Number(20)  |                                 | Tender amount.                                                                                   | Yes              |

| Record Name | Field Name                       | Field Type  | Default Value | Description                                                                                                                                                                                                                                                                                | Required |
|-------------|----------------------------------|-------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|
|             | Credit Card Number               | Numeric(40) |               |                                                                                                                                                                                                                                                                                            | No       |
|             | Credit Card Expiration Date      | Numeric(8)  |               | Format YYYYMMDD                                                                                                                                                                                                                                                                            | No       |
|             | Credit Card Authorization Number | Char(16)    |               |                                                                                                                                                                                                                                                                                            | No       |
|             | Credit Card Authorization Source | Char(6)     |               | Contains whether the authorization number was electronically transmitted or manually keyed in after obtaining it via a telephone call. The code type for this field is 'CCAS'.                                                                                                             | No       |
|             | Credit Card Entry Mode           | Char(6)     |               | Contains the method in which the transaction was entered at the POS. Possible entry modes could include: Terminal Used, Magnetic Strip Track One Read, Magnetic Strip Two Read, Magnetic Strip One Transmitted, or Magnetic Strip Two Transmitted. The code type for this field is 'CCEM'. | No       |

| Record Name | Field Name                          | Field Type | Default Value | Description                                                                                                                                                                                                                                              | Required                             |
|-------------|-------------------------------------|------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------|
|             | Credit Card Cardholder Verification | Char(6)    |               | Contains the method of identification that was used by the cardholder to verify their identity. Possible values include Signature Verified ('S'), Card Shown ('C'), PIN Entered ('P'), Mail Order / Phone ('M'). The code type for this field is 'CCVF'. | No                                   |
|             | Credit Card Terminal ID             | Char(5)    |               | Contains the identification code of the terminal within the store that the transaction was transmitted.                                                                                                                                                  | No                                   |
|             | Credit Card Special Conditions      | Char(6)    |               | Contains the special condition of the transaction (mail, phone or electronic-secured or non-secured authentication). The code type for this field is 'CCSC'.                                                                                             | No                                   |
|             | Voucher Number                      | Char(25)   |               |                                                                                                                                                                                                                                                          | No                                   |
|             | Voucher Age                         | Numeric(5) |               | Age of the gift certificate. redeemed date minus sold date.                                                                                                                                                                                              | Yes if Tender Type Group is 'VOUCH'. |
|             | Escheat Date                        | Numeric(8) |               | Date on which this gift certificate escheats. Format is YYYYMMDD.                                                                                                                                                                                        | Yes if voucher can escheat.          |

| Record Name  | Field Name                  | Field Type | Default Value                | Description                                                                                 | Required                                   |
|--------------|-----------------------------|------------|------------------------------|---------------------------------------------------------------------------------------------|--------------------------------------------|
|              | Coupon Number               | Char(16)   |                              |                                                                                             | Yes if Tender Type Group is 'COUPON'.      |
|              | Coupon Reference Number     | Char(16)   |                              |                                                                                             | No. Only if Tender Type Group is 'COUPON'. |
| File Trailer | File Type Record Descriptor | Char(5)    | FTAIL                        | Identifies file record type                                                                 |                                            |
|              | File Line Identifier        | Number(10) | specified by external system | ID of current line being processed by input file.                                           | Yes                                        |
|              | File Record Counter         | Number(10) |                              | Number of records/transactions processed in current file (only records between head & tail) | Yes                                        |

## RDW Form of Payment File after translation by resa2rdw

| Record Name | Field Name       | Field Type  | Default Value                | Description                                                                     | Required |
|-------------|------------------|-------------|------------------------------|---------------------------------------------------------------------------------|----------|
|             | Business date    | Numeric(8)  |                              | Format YYYYMMDD                                                                 | Yes      |
|             | Transaction Date | Numeric(14) | transaction date             | Date sale/return transaction was processed at the POS. Format YYYYMMDDH H24MISS | Yes      |
|             | Location         | Number(10)  | specified by external system | Store or warehouse identifier.                                                  | Yes      |



| Record Name | Field Name                       | Field Type  | Default Value                    | Description                                                                                      | Required         |
|-------------|----------------------------------|-------------|----------------------------------|--------------------------------------------------------------------------------------------------|------------------|
|             | Cashier Identifier               | Char(10)    |                                  | The cashier number. This is the unique employee number.                                          | Yes, -1 for null |
|             | Register Identifier              | Char(5)     |                                  |                                                                                                  | Yes, -1 for null |
|             | Sales Sign                       | Char(1)     | 'P' - positive<br>'N' – negative | Determines if the Total Sales Quantity and Total Sales Value are positive or negative.           | Yes              |
|             | Transaction Sequence Number      | Numeric(20) |                                  | Unique reference used within sales audit to represent the date/store/register/transaction number | Yes              |
|             | Revision number                  | Number(3)   |                                  | Number used to identify the version of the transaction being sent.                               | Yes              |
|             | Transaction Type                 | Char(6)     |                                  | Transaction type code.                                                                           | Yes              |
|             | Tender type group                | Char(6)     |                                  |                                                                                                  | Yes              |
|             | Tender type id                   | Numeric(6)  |                                  | Tender type code.                                                                                | Yes              |
|             | Tender amount                    | Number(20)  |                                  | Tender amount.                                                                                   | Yes              |
|             | Credit Card Number               | Numeric(40) |                                  |                                                                                                  | No               |
|             | Credit Card Expiration Date      | Numeric(8)  |                                  | Format YYYYMMDD                                                                                  | No               |
|             | Credit Card Authorization Number | Char(16)    |                                  |                                                                                                  | No               |

| Record Name | Field Name                          | Field Type | Default Value | Description                                                                                                                                                                                                                                                                                | Required |
|-------------|-------------------------------------|------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|
|             | Credit Card Authorization Source    | Char(6)    |               | Contains whether the authorization number was electronically transmitted or manually keyed in after obtaining it via a telephone call. The code type for this field is 'CCAS'.                                                                                                             | No       |
|             | Credit Card Entry Mode              | Char(6)    |               | Contains the method in which the transaction was entered at the POS. Possible entry modes could include: Terminal Used, Magnetic Strip Track One Read, Magnetic Strip Two Read, Magnetic Strip One Transmitted, or Magnetic Strip Two Transmitted. The code type for this field is 'CCEM'. | No       |
|             | Credit Card Cardholder Verification | Char(6)    |               | Contains the method of identification that was used by the cardholder to verify their identity. Possible values include Signature Verified ('S'), Card Shown ('C'), PIN Entered ('P'), Mail Order / Phone ('M'). The code type for this field is 'CCVF'.                                   | No       |

| Record Name | Field Name                     | Field Type | Default Value | Description                                                                                                                                                  | Required                                   |
|-------------|--------------------------------|------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------|
|             | Credit Card Terminal ID        | Char(5)    |               | Contains the identification code of the terminal within the store that the transaction was transmitted.                                                      | No                                         |
|             | Credit Card Special Conditions | Char(6)    |               | Contains the special condition of the transaction (mail, phone or electronic-secured or non-secured authentication). The code type for this field is 'CCSC'. | No                                         |
|             | Voucher Number                 | Char(25)   |               |                                                                                                                                                              | No                                         |
|             | Voucher Age                    | Numeric(5) |               | Age of the gift certificate. redeemed date minus sold date.                                                                                                  | Yes if Tender Type Group is 'VOUCH'.       |
|             | Escheat Date                   | Numeric(8) |               | Date on which this gift certificate escheats. Format is YYYYMMDD.                                                                                            | Yes if voucher can escheat.                |
|             | Coupon Number                  | Char(16)   |               |                                                                                                                                                              | Yes if Tender Type Group is 'COUPON'.      |
|             | Coupon Reference Number        | Char(16)   |               |                                                                                                                                                              | No. Only if Tender Type Group is 'COUPON'. |

### Store Totals Information

| Record Name | Field Name                  | Field Type | Default Value | Description                 | Required |
|-------------|-----------------------------|------------|---------------|-----------------------------|----------|
| File Header | File Type Record Descriptor | Char(5)    | FHEAD         | Identifies file record type |          |

| Record Name | Field Name                  | Field Type  | Default Value                    | Description                                                                            | Required |
|-------------|-----------------------------|-------------|----------------------------------|----------------------------------------------------------------------------------------|----------|
|             | File Line Identifier        | Number(10)  | specified by external system     | ID of current line being processed by input file.                                      | Yes      |
|             | File Type Definition        | Char(4)     | RDWS                             | Identifies file as 'RDW Store Totals file'                                             | Yes      |
|             | File Create Date            | Numeric(14) | create date                      | date file was written by external system. Format YYYYMMDDH H24MISS                     | Yes      |
| File Detail | File Type Record Descriptor | Char(5)     | FDETL                            | Identifies transaction record type                                                     |          |
|             | File Line Identifier        | Number(10)  | specified by external system     | ID of current line being processed by input file.                                      | Yes      |
|             | Business date               | Number(8)   |                                  | Format YYYYMMDD                                                                        | Yes      |
|             | Location                    | Number(10)  | specified by external system     | Store or warehouse identifier                                                          | Yes      |
|             | Sales Sign                  | Char(1)     | 'P' - positive<br>'N' - negative | Determines if the Total Sales Quantity and Total Sales Value are positive or negative. | Yes      |
|             | Total ID                    | Char(10)    |                                  | Category identifier used to determine the type of total.                               | Yes      |
|             | Reference Number 1          | Char(30)    |                                  |                                                                                        | No       |
|             | Reference Number 2          | Char(30)    |                                  |                                                                                        | No       |
|             | Reference Number 3          | Char(30)    |                                  |                                                                                        | No       |
|             | Total Sign                  | Char(1)     | 'P' - positive<br>'N' - negative |                                                                                        | Yes      |

| Record Name  | Field Name                  | Field Type | Default Value                | Description                                                                                 | Required |
|--------------|-----------------------------|------------|------------------------------|---------------------------------------------------------------------------------------------|----------|
|              | Total Amount                | Number(20) |                              | Total over/short amount with 4 implied decimal places.                                      | Yes      |
| File Trailer | File Type Record Descriptor | Char(5)    | FTAIL                        | Identifies file record type                                                                 |          |
|              | File Line Identifier        | Number(10) | specified by external system | ID of current line being processed by input file.                                           | Yes      |
|              | File Record Counter         | Number(10) |                              | Number of records/transactions processed in current file (only records between head & tail) | Yes      |

#### Store Totals Information after Translation by resa2rdw

| Record Name | Field Name         | Field Type | Default Value                   | Description                                                                            | Required |
|-------------|--------------------|------------|---------------------------------|----------------------------------------------------------------------------------------|----------|
|             | Business date      | Number(8)  |                                 | Format YYYYMMDD                                                                        | Yes      |
|             | Location           | Number(10) | specified by external system    | Store or warehouse identifier                                                          | Yes      |
|             | Sales Sign         | Char(1)    | 'P'- positive<br>'N' – negative | Determines if the Total Sales Quantity and Total Sales Value are positive or negative. | Yes      |
|             | Total ID           | Char(10)   |                                 | Category identifier used to determine the type of total.                               | Yes      |
|             | Reference Number 1 | Char(30)   |                                 |                                                                                        | No       |
|             | Reference Number 2 | Char(30)   |                                 |                                                                                        | No       |
|             | Reference Number 3 | Char(30)   |                                 |                                                                                        | No       |

| Record Name | Field Name   | Field Type | Default Value                    | Description                                            | Required |
|-------------|--------------|------------|----------------------------------|--------------------------------------------------------|----------|
|             | Total Sign   | Char(1)    | 'P' - positive<br>'N' - negative |                                                        | Yes      |
|             | Total Amount | Number(20) |                                  | Total over/short amount with 4 implied decimal places. | Yes      |

### Cashier/Register Totals Information

| Record Name | Field Name                  | Field Type  | Default Value                | Description                                                        | Required |
|-------------|-----------------------------|-------------|------------------------------|--------------------------------------------------------------------|----------|
| File Header | File Type Record Descriptor | Char(5)     | FHEAD                        | Identifies file record type                                        |          |
|             | File Line Identifier        | Number(10)  | specified by external system | ID of current line being processed by input file.                  | Yes      |
|             | File Type Definition        | Char(4)     | RDWC                         | Identifies file as 'RDW Cashier/Register Totals file'              | Yes      |
|             | File Create Date            | Numeric(14) | create date                  | date file was written by external system. Format YYYYMMDDH H24MISS | Yes      |
| File Detail | File Type Record Descriptor | Char(5)     | FDETL                        | Identifies transaction record type                                 |          |
|             | File Line Identifier        | Number(10)  | specified by external system | ID of current line being processed by input file.                  | Yes      |
|             | Business date               | Number(8)   |                              | Format YYYYMMDD                                                    | Yes      |
|             | Location                    | Number(10)  | specified by external system | Store or warehouse identifier                                      | Yes      |

| Record Name | Field Name         | Field Type | Default Value                    | Description                                                                            | Required                                                                                                                |
|-------------|--------------------|------------|----------------------------------|----------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
|             | Cashier Identifier | Char(10)   |                                  | The cashier number                                                                     | If Cashier_id is NULL then Register_id has value. If Cashier_id has value then Register_id is NULL.<br>Yes, -1 for null |
|             | Register ID        | Char(5)    |                                  | The register identifier                                                                | If Cashier_id is NULL then Register_id has value. If Cashier_id has value then Register_id is NULL.<br>Yes, -1 for null |
|             | Sales Sign         | Char(1)    | 'P' - positive<br>'N' - negative | Determines if the Total Sales Quantity and Total Sales Value are positive or negative. | Yes                                                                                                                     |
|             | Total ID           | Char(10)   |                                  | Category identifier used to determine the type of total.                               | Yes                                                                                                                     |
|             | Reference Number 1 | Char(30)   |                                  |                                                                                        | No                                                                                                                      |
|             | Reference Number 2 | Char(30)   |                                  |                                                                                        | No                                                                                                                      |
|             | Reference Number 3 | Char(30)   |                                  |                                                                                        | No                                                                                                                      |
|             | Total Sign         | Char(1)    | 'P' - positive<br>'N' - negative |                                                                                        | Yes                                                                                                                     |
|             | Total Amount       | Number(20) |                                  | Total over/short amount with 4 implied decimal places.                                 | Yes                                                                                                                     |

| Record Name  | Field Name                  | Field Type | Default Value                | Description                                                                                 | Required |
|--------------|-----------------------------|------------|------------------------------|---------------------------------------------------------------------------------------------|----------|
| File Trailer | File Type Record Descriptor | Char(5)    | FTAIL                        | Identifies file record type                                                                 |          |
|              | File Line Identifier        | Number(10) | specified by external system | ID of current line being processed by input file.                                           | Yes      |
|              | File Record Counter         | Number(10) |                              | Number of records/transactions processed in current file (only records between head & tail) | Yes      |

**Cashier/ Register Totals Information after Translation by resa2rdw**

| Record Name | Field Name         | Field Type | Default Value                | Description                   | Required                                                                                                                |
|-------------|--------------------|------------|------------------------------|-------------------------------|-------------------------------------------------------------------------------------------------------------------------|
|             | Business date      | Number(8)  |                              | Format YYYYMMDD               | Yes                                                                                                                     |
|             | Location           | Number(10) | specified by external system | Store or warehouse identifier | Yes                                                                                                                     |
|             | Cashier Identifier | Char(10)   |                              | The cashier number            | If Cashier_id is NULL then Register_id has value. If Cashier_id has value then Register_id is NULL.<br>Yes, -1 for null |
|             | Register ID        | Char(5)    |                              | The register identifier       | If Cashier_id is NULL then Register_id has value. If Cashier_id has value then Register_id is NULL.<br>Yes, -1 for null |



| Record Name | Field Name         | Field Type | Default Value                    | Description                                                                            | Required |
|-------------|--------------------|------------|----------------------------------|----------------------------------------------------------------------------------------|----------|
|             | Sales Sign         | Char(1)    | 'P' - positive<br>'N' - negative | Determines if the Total Sales Quantity and Total Sales Value are positive or negative. | Yes      |
|             | Total ID           | Char(10)   |                                  | Category identifier used to determine the type of total.                               | Yes      |
|             | Reference Number 1 | Char(30)   |                                  |                                                                                        | No       |
|             | Reference Number 2 | Char(30)   |                                  |                                                                                        | No       |
|             | Reference Number 3 | Char(30)   |                                  |                                                                                        | No       |
|             | Total Sign         | Char(1)    | 'P' - positive<br>'N' - negative |                                                                                        | Yes      |
|             | Total Amount       | Number(20) |                                  | Total over/short amount with 4 implied decimal places.                                 | Yes      |

## saexprms (Sales Audit Export to RMS)

### Functional Area

Oracle Retail Sales Audit (ReSA)

### Module Affected

SAEXPRMS.PC

### Design Overview

The purpose of this batch module is to fetch all corrected sale and return transactions that do not have RMS errors from the Oracle Retail Sales Audit (ReSA) database tables for transmission to the Oracle Retail Merchandising System (RMS). If

sa\_system\_options.unit\_of\_work is 'S', then the whole store/day is skipped if any RMS error is found. If this value is 'T', then only transactions with RMS errors are skipped.

If the transaction has a status of Deleted and it has previously been transmitted, a reversal of the transaction is sent.

A file of type POSU is generated for each store/day.

## Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                    |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Any time. The processing cycle of the Sales Audit is a 24/7 system.                                                                                                            |
| Scheduling Considerations | This program should run towards the end of the Sales Auditing cycle where the total (SATOTALS.PC) and rule (SARULES.PC) data are ready to be exported to the external systems. |
| Pre-Processing            | saprepost saexprms pre                                                                                                                                                         |
| Post-Processing           | saprepost saexprms post                                                                                                                                                        |
| Threading Scheme          | Multi-threaded by store                                                                                                                                                        |

## Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records are fetched, updated and inserted in batches of `pl_commit_max_ctr`. Only two commits are done, one to establish the store/day lock and another at the end, to release the lock after a store/day is completely processed. The POSU formatted output file is created with a temporary name and renamed just before the end of store/day commit.

In case of failure, all work done is rolled back to the point right after the call to `get_lock()` and releases the lock. Thus, the rollback segment should be large enough to hold all inserts into `sa_exported` for one store/day.

## Locking Strategy

This program is attempted to obtain a read lock on the store/day with a call to `get_lock()` library function. If this fails, go on to the next store/day and log the problem to the error log. If the previous store day transaction failed, then release its lock with a call to `release_lock()` and rollback the transaction

## Security Considerations

Sales Audit for all stores are stored in a Unix file with the processes default permissions (umask). Care should be exercised so that this file cannot be tampered with.

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| SA_STORE_DAY    | Yes    | No     | No     | No     |
| SA_EXPORT_LOG   | Yes    | No     | Yes    | No     |
| V_RESTART_STORE | Yes    | No     | No     | No     |
| STORE           | Yes    | No     | No     | No     |
| CURRENCIES      | Yes    | No     | No     | No     |

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| SA_TRAN_HEAD           | Yes    | No     | No     | No     |
| SA_ERROR               | Yes    | No     | No     | No     |
| SA_ERROR_IMPACT        | Yes    | No     | No     | No     |
| SA_EXPORTED            | Yes    | Yes    | No     | No     |
| SA_TRAN_SEQ_TEMP       | No     | Yes    | No     | Yes    |
| SA_TRAN_HEAD_REV       | Yes    | No     | No     | No     |
| SA_EXPORTED_REV        | Yes    | No     | No     | No     |
| SA_SYSTEM_OPTIONS      | Yes    | No     | No     | No     |
| SA_TRAN_ITEM_REV       | Yes    | No     | No     | No     |
| ITEM_MASTER            | Yes    | No     | No     | No     |
| SA_TRAN_DISC_REV       | Yes    | No     | No     | No     |
| SA_TRAN_DISC           | Yes    | No     | No     | No     |
| SA_TRAN_ITEM           | Yes    | No     | No     | No     |
| SA_TRAN_SEQ_TEMP       | Yes    | Yes    | No     | Yes    |
| SA_STORE_DAY_READ_LOCK | No     | Yes    | No     | Yes    |

## I/O Specification

### Output File Layout

| Record Name | Field Name            | Field Type | Default Value | Description                                                                                                              |
|-------------|-----------------------|------------|---------------|--------------------------------------------------------------------------------------------------------------------------|
| FHEAD       | Record descriptor     | Char(5)    | FHEAD         | Identifies the file record type                                                                                          |
|             | File Line Id          | Char(10)   | 0000000001    | Sequential file line number                                                                                              |
|             | File type definition  | Char(4)    | POSU          | Identifies the file type                                                                                                 |
|             | File Create Date      | Char(14)   |               | File Create Date in YYYYMMDDHHMMSS format                                                                                |
|             | Store                 | Number(10) |               | Store location                                                                                                           |
|             | Vat include indicator | Char(1)    |               | Determines whether or not the store stores values including vat. Not required but populated by Oracle Retail sales audit |

| Record Name | Field Name               | Field Type | Default Value | Description                                                                                                                  |
|-------------|--------------------------|------------|---------------|------------------------------------------------------------------------------------------------------------------------------|
|             | Vat region               | Number(4)  |               | Vat region the given location is in. Not required but populated by Oracle Retail sales audit                                 |
|             | Currency code            | Char(3)    |               | Currency of the given location. Not required but populated by Oracle Retail sales audit                                      |
|             | Currency retail decimals | Number(1)  |               | Number of decimals supported by given currency for retails. Not required but populated by Oracle Retail sales audit          |
| THEAD       | Record descriptor        | Char(5)    | THEAD         | Identifies the file record type                                                                                              |
|             | File Line Id             | Char(10)   |               | Sequential file line number                                                                                                  |
|             | Transaction date         | Char(14)   |               | Transaction date in YYYYMMDDHHMMSS format. Corresponds to the date that the sale/return transaction was processed at the POS |
|             | Item Type                | Char(3)    | REF or ITM    | Can be REF or ITM                                                                                                            |
|             | Item                     | Char(25)   |               | Id number of the ITM or REF                                                                                                  |
|             | Dept                     | Number(4)  |               | Department of item sold or returned.                                                                                         |
|             | Class                    | Number(4)  |               | Class of item sold or returned.                                                                                              |
|             | Sub Class                | Number(4)  |               | Subclass of item sold or returned.                                                                                           |
|             | Pack Ind                 | Char(1)    |               | Pack indicator of item sold or returned.                                                                                     |
|             | Item Level               | Number(1)  |               | Item level of item sold or returned.                                                                                         |
|             | Tran level               | Number(1)  |               | Transaction level of item sold or returned.                                                                                  |
|             | Wastage Type             | Char(6)    |               | Wastage type of item sold or returned                                                                                        |
|             | Wastage pct              | Number(12) |               | Waste pct (4 implied decimal places)                                                                                         |

| Record Name | Field Name              | Field Type | Default Value | Description                                                                            |
|-------------|-------------------------|------------|---------------|----------------------------------------------------------------------------------------|
|             | Tran type               | Char(1)    |               | Transaction type code to specify whether transaction is a sale or a return             |
|             | Drop Shipment indicator | Char(1)    |               | Indicates whether the transaction is a drop shipment or not.                           |
|             | Total sales qty         | Number(12) |               | Total sales quantity (4 implied decimal places)                                        |
|             | Selling UOM             | Char(4)    |               | Selling Unit of Measure for the item                                                   |
|             |                         |            |               |                                                                                        |
|             | Sales sign              | Char(1)    |               | Determines if the Total Sales Quantity and Total Sales Value are positive or negative. |
|             | Total Sales Value       | Number(20) |               | Total sales value of goods sold/returned (4 implied decimal places)                    |
|             | Last Date time modified | Char(14)   |               | Date and time of last modification in YYYYMMDDHHMMSS format. For VBO future use        |
|             | Catchweight indicator   | Char(1)    |               | Indicates if item is a catchweight item.                                               |
|             | Total weight            | Number(12) |               | The actual weight of the item, only populated if catchweight_ind = 'Y'                 |
|             | Sub Tran type indicator | Char(1)    |               | Tran type for ReSA<br>Valid values are 'A', 'D', NULL                                  |
| TDETL       | Record descriptor       | Char(5)    | TDETL         | Identifies the file record type                                                        |
|             | File Line Id            | Char(10)   |               | Sequential file line number                                                            |
|             | Promo Tran Type         | Char(6)    |               | Code for the promotional type from code_detail where code_type = 'PRMT'                |
|             | Promotion Number        | Number(10) |               | Promotion number from RMS                                                              |

| Record Name | Field Name          | Field Type | Default Value | Description                                                                                 |
|-------------|---------------------|------------|---------------|---------------------------------------------------------------------------------------------|
|             | Sales quantity      | Number(12) |               | Sales quantity sold for this promotion type (4 implied decimal places)                      |
|             | Sales value         | Number(20) |               | Sales value for this promotion type (4 implied decimal places)                              |
|             | Discount value      | Number(20) |               | Discount value for this promotion type (4 implied decimal places)                           |
|             | Promotion component | Number(10) |               | Links the promotion to additional pricing attributes                                        |
| TTAIL       | Record descriptor   | Char(5)    | TTAIL         | Identifies the file record type                                                             |
|             | File Line Id        | Char(10)   |               | Sequential file line number                                                                 |
|             | Tran Record Counter | Number(6)  |               | Number of TDETL records in this transaction set                                             |
| FTAIL       | Record descriptor   | Char(5)    | FTAIL         | Identifies the file record type                                                             |
|             | File Line Id        | Number(10) |               | Sequential file line number                                                                 |
|             | File Record counter | Number(10) |               | Number of records/transactions processed in current file (only records between head & tail) |

## saexpuar (Universal Account Reconciliation System Export)

### Functional Area

Oracle Retail Sales Audit (ReSA)

### Module Affected

SAEXPUAR.PC

### Design Overview

The SAEXPUAR program is used to select the lottery, bank deposit, money order and credit card totals and writes them to output files for export to the 'J. Driscoll & Associates' UAR application. For each store day, SAEXPUAR posts specified totals to their appropriate output files.

## Scheduling Constraints

| Schedule Information      | Description                                                                      |
|---------------------------|----------------------------------------------------------------------------------|
| Processing Cycle          | Any time. The processing cycle of the Sales Audit is a 24/7 system.              |
| Scheduling Considerations | This program should run after the ReSA Totaling process and Audit Rules process. |
| Pre-Processing            | N/A                                                                              |
| Post-Processing           | N/A                                                                              |
| Threading Scheme          | N/A                                                                              |

## Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records are fetched, updated and inserted in batches of `commit_max_ctr`. Only two commits are done. One to establish the store/day lock (this is done by the package) and the other is done at the end, after a store/day is completely processed.

## Locking Strategy

This program is attempted to obtain a read lock on the store/day with a call to `get_lock`. If this fails, move on to the next store/day and log the problem to the error log.

## Security Considerations

Sales Audit for all the stores is stored in a Unix file with processes and default permissions (`umask`). Care should be taken to avoid tampering of this file.

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| SA_STORE_DAY    | Yes    | No     | No     | No     |
| SA_EXPORT_LOG   | Yes    | No     | Yes    | No     |
| SA_EXPORTED     | No     | Yes    | Yes    | No     |
| SA_EXPORTED_REV | Yes    | No     | No     | No     |
| SA_TOTAL        | Yes    | No     | No     | No     |
| SA_TOTAL_HEAD   | Yes    | No     | No     | No     |
| SA_HQ_VALUE     | Yes    | No     | No     | No     |
| SA_STORE_VALUE  | Yes    | No     | No     | No     |
| SA_SYS_VALUE    | Yes    | No     | No     | No     |
| SA_POS_VALUE    | Yes    | No     | No     | No     |

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| SA_TOTAL_USAGE          | Yes    | No     | No     | No     |
| SA_STORE_DAY_WRITE_LOCK | Yes    | No     | No     | No     |
| SA_STORE_DAY_READ_LOCK  | Yes    | Yes    | No     | Yes    |

## I/O Specification

### Output File Layout

The output file contains one line for each store/day detail record in a comma-delimited format. The fields are surrounded by double quotes. For example, a record for store 1000 on May 20, 2001 with an amount of 19.99 looks something like this:

"1", "1000", "1999", "20010520", "2", "", "1", "", "", "", "", "", "", "", "", "", "MN", "RET"

| Field Name            | Field Type | Description                                                         |
|-----------------------|------------|---------------------------------------------------------------------|
| Detail Flag           | Char       | "1" for detail record                                               |
| Store                 | Number     | Store Number                                                        |
| Amount                | Number     | Total Value * 100 (with 2 implied decimal places).                  |
| TranDate              | Char       | Transaction Date in 'YYYYMMDD' format                               |
| UAR TranCode          | Char       | Transaction Code. "1" for negative amount, "2" for positive amount. |
| User Defined Value 1  | Char       | Ref No 1 on SA_TOTAL.                                               |
| User Defined Value 2  | Char       | Total Seq No. on SA_TOTAL                                           |
| User Defined Value 3  | Char       | Ref No 2 on SA_TOTAL                                                |
| User Defined Value 4  | Char       | Ref No. 3 on SA_TOTAL                                               |
| User Defined Value 5  | Char       | Not used                                                            |
| User Defined Value_6  | Char       | Not used                                                            |
| User Defined Value 7  | Char       | Not used                                                            |
| User Defined Value 8  | Char       | Not used                                                            |
| User Defined Value 9  | Char       | Not used                                                            |
| User Defined Value 10 | Char       | Not used                                                            |
| State                 | Char       | State                                                               |
| Account               | Char       | Total Identification on SA_TOTAL                                    |



## sagetref (Sales Audit Get Reference)

### Functional Area

Oracle Retail Sales Audit (ReSA)

### Module Affected

SAGETREF.PC

### Design Overview

This program fetches all reference information needed by SAIMPTLOG.PC and writes this information out to separate output files. One file contains a listing of all items in the system. A second file contains information about all items that have wastage associated with them. A third file contains reference items. A fourth file contains primary variant information. A fifth file contains all variable weight UPC definitions in the system. A sixth file contains all of the valid store/day combinations in the system. A seventh file contains all code types and codes used in field level validation. An eighth file contains all error codes, error descriptions and systems affected by the error. A ninth file contains the credit card validation mappings. A tenth file contains the store\_pos mappings. An eleventh file contains the tender type mappings. A twelfth file contains the merchant code mappings. A thirteenth file contains the partner mappings. A fourteenth file contains the supplier mappings. A fifteenth file contains employee mappings. Finally a sixteenth file contains banner information. These files are used by the automated audit to validate information without repeatedly hitting the database.

### Scheduling Constraints

| Schedule Information      | Description                                                                                        |
|---------------------------|----------------------------------------------------------------------------------------------------|
| Processing Cycle          | Daily – Anytime – Sales Audit is a 24/7 system.                                                    |
| Scheduling Considerations | This module should be executed in the earliest phase, before the first import of RTLOGs into ReSA. |
| Pre-Processing            | SASTDYCR.PC                                                                                        |
| Post-Processing           | SAIMPTLOG.PC                                                                                       |
| Threading Scheme          | N/A                                                                                                |

### Restart/Recovery

N/A

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| ITEM_MASTER          | Yes    | No     | No     | No     |
| ITEM_LOC             | Yes    | No     | No     | No     |
| VAR_UPC_EAN          | Yes    | No     | No     | No     |
| SA_STORE_DAY         | Yes    | No     | No     | No     |
| SA_STORE             | Yes    | No     | No     | No     |
| SA_IMPORT_LOG        | Yes    | No     | No     | No     |
| CURRENCIES           | Yes    | No     | No     | No     |
| ADDR                 | Yes    | No     | No     | No     |
| CODE_DETAIL          | Yes    | No     | No     | No     |
| SA_ERROR_CODES       | Yes    | No     | No     | No     |
| SA_CC_VAL            | Yes    | No     | No     | No     |
| SA_STORE_POS         | Yes    | No     | No     | No     |
| POS_TENDER_TYPE_HEAD | Yes    | No     | No     | No     |
| NON_MERCH_CODE_HEAD  | Yes    | No     | No     | No     |
| PARTNER              | Yes    | No     | No     | No     |
| SUPS                 | Yes    | No     | No     | No     |
| SA_STORE_EMP         | Yes    | No     | No     | No     |
| STORE                | Yes    | No     | No     | No     |
| BANNER               | Yes    | No     | No     | No     |
| CHANNELS             | Yes    | No     | No     | No     |

**I/O Specification****Output File Layout****File Name: Item File**

| Record Name | Field Name      | Field Type | Default Value | Description              |
|-------------|-----------------|------------|---------------|--------------------------|
|             | Item            | Char(25)   |               | Item number              |
|             | Dept            | Number(4)  |               | Department id            |
|             | Class           | Number(4)  |               | Class id                 |
|             | Subclass        | Number(4)  |               | Subclass id              |
|             | Standard UOM    | Char(4)    |               | Standard Unit of Measure |
|             | Catchweight Ind | Char(1)    |               | Catch weight indicator   |

**File Name: Waste data File**

| Record Name | Field Name | Field Type   | Default Value | Description |
|-------------|------------|--------------|---------------|-------------|
|             | Item       | Char(25)     |               | Item number |
|             | Waste type | Char(6)      |               | Waste type  |
|             | Waste pct  | Number(12,4) |               | Waste pct   |

**File Name: Reference Item data**

| Record Name | Field Name | Field Type | Default Value | Description           |
|-------------|------------|------------|---------------|-----------------------|
|             | Ref Item   | Char(25)   |               | Reference Item number |
|             | Item       | Char(25)   |               | Item number           |

**File Name: Primary variant data file**

| Record Name | Field Name   | Field Type | Default Value | Description     |
|-------------|--------------|------------|---------------|-----------------|
|             | Location     | Number(10) |               | Location number |
|             | Item         | Char(25)   |               | Item number     |
|             | Prim Variant | Char(25)   |               | Primary variant |

**File Name: Variable Weight UPC definition file**

| Record Name | Field Name       | Field Type | Default Value | Description        |
|-------------|------------------|------------|---------------|--------------------|
|             | Format Id        | Char(1)    |               | Format id          |
|             | Format desc      | Char(20)   |               | Format description |
|             | Prefix length    | Number(1)  |               | Pefix Length       |
|             | Begin item digit | Number(2)  |               | Item digit begin   |
|             | Begin war digit  | Number(2)  |               | War digit begin    |
|             | Check digit      | Number(2)  |               | Check digit        |
|             | Default prefix   | Number(1)  |               | Default prefix     |
|             | Prefix           | Number(1)  |               | Prefix             |

**File Name: Valid Store/Day Combination file**

| Record Name | Field Name        | Field Type | Default Value | Description                               |
|-------------|-------------------|------------|---------------|-------------------------------------------|
|             | Store             | Number(10) |               | Store number                              |
|             | Business date     | Char(8)    |               | Buisness date in YYYYMMDD format          |
|             | Store day seq no  | Number(20) |               | Store day sequence number                 |
|             | Day               | Number(3)  |               | Day                                       |
|             | Tran no generated | Char(6)    |               | Generated transaction number              |
|             | System code       | Char(1)    |               | If system_code is 'POS' then 'Y' else 'N' |
|             | Currency rtl dec  | Number(1)  |               | Currency rtl dec                          |
|             | Currency code     | Char(3)    |               | Currency code                             |
|             | Country id        | Char(3)    |               | Country id                                |

**File Name: Codes file**

| Record Name | Field Name | Field Type | Default Value | Description   |
|-------------|------------|------------|---------------|---------------|
|             | Code type  | Char(4)    |               | Code type     |
|             | Code       | Char(6)    |               | Code id       |
|             | Code seq   | Number(4)  |               | Code sequence |

**File Name: Error Information File**

| Record Name | Field Name   | Field Type | Default Value | Description            |
|-------------|--------------|------------|---------------|------------------------|
|             | Error code   | Char(25)   |               | Error code             |
|             | Error desc   | Char(255)  |               | Error description      |
|             | Rec solution | Char(255)  |               | Error rectify solution |

**File Name: Credit Card Validation Mapping File**

| Record Name | Field Name     | Field Type | Default Value | Description     |
|-------------|----------------|------------|---------------|-----------------|
|             | Length         | Number(2)  |               | Length          |
|             | From Prefix    | Number(6)  |               | From Prefix     |
|             | To Prefix      | Number(6)  |               | To Prefix       |
|             | Tender type id | Number(6)  |               | Tender type id  |
|             | VAL Type       | Char(6)    | NONE          | Validation Type |

**File Name: Store POS Mapping File**

| Record Name | Field Name     | Field Type | Default Value | Description              |
|-------------|----------------|------------|---------------|--------------------------|
|             | Store          | Number(10) |               | Store                    |
|             | POS Type       | Char(6)    |               | Point Of Sale type       |
|             | Start Tran No. | Number(10) |               | Start transaction number |
|             | End Tran No.   | Number(10) |               | End transaction number   |

**File Name: Tender Type Mapping File**

| Record Name | Field Name | Field Type | Default Value | Description             |
|-------------|------------|------------|---------------|-------------------------|
|             | Group      | Char(6)    |               | Tender type Group       |
|             | Id         | Number(6)  |               | Tender type id          |
|             | Desc       | Char(120)  |               | Tender type description |

**File Name: Merchant Code Mapping File**

| Record Name | Field Name     | Field Type | Default Value | Description       |
|-------------|----------------|------------|---------------|-------------------|
|             | Non Merch Code | Char(6)    |               | Non merchant code |

**File Name: Partner Mapping File**

| Record Name | Field Name   | Field Type | Default Value | Description  |
|-------------|--------------|------------|---------------|--------------|
|             | Partner Type | Char(6)    |               | Partner Type |
|             | Partner Id   | Char(10)   |               | Partner Id   |

**File Name: Supplier Mapping File**

| Record Name | Field Name | Field Type | Default Value | Description     |
|-------------|------------|------------|---------------|-----------------|
|             | Supplier   | Number(10) |               | Supplier Id     |
|             | Sup status | Char(1)    |               | Supplier status |

**File Name: Employee Mapping File**

| Record Name | Field Name | Field Type | Default Value | Description      |
|-------------|------------|------------|---------------|------------------|
|             | Store      | Number(10) |               | Store Id         |
|             | POS Id     | Char(10)   |               | Point Of Sale Id |
|             | Emp Id     | Char(10)   |               | Employee Id      |

**File Name: Banner Information File**

| Record Name | Field Name  | Field Type | Default Value | Description |
|-------------|-------------|------------|---------------|-------------|
|             | Store       | Number(10) |               | Store Id    |
|             | Banner data | Number(4)  |               | Banner Id   |

## saimpadj (Sales Audit Import Adjustments)

### Functional Area

Oracle Retail Sales Audit (ReSA)

### Module Affected

SAIMPADJ.PC

### Design Overview

This module posts external system adjustments to the Sales Audit total value table.

The sales audit adjustments are passed to the module in an external file.

Records that fail necessary validations would be written to the reject file. The input and reject file names are passed as arguments.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                  |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Anytime - Sales Audit is a 24/7 system                                                                                                       |
| Scheduling Considerations | This module should be executed after the ReSA transaction import process (SAIMPTLOG.PC), and before the ReSA totaling process (SATOTALS.PC). |
| Pre-Processing            | N/A                                                                                                                                          |
| Post-Processing           | N/A                                                                                                                                          |
| Threading Scheme          | N/A                                                                                                                                          |

### Restart/Recovery

Restart/recovery logic for file based processing is used. The logical unit of work for this module is a parameterized number defined in restart tables.

### Locking Strategy

Record level locking is done on sa\_store\_day before updating.

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| SA_TOTAL       | Yes    | No     | No     | No     |
| SA_HQ_VALUE    | No     | Yes    | No     | No     |
| SA_STORE_DAY   | Yes    | No     | Yes    | No     |
| SA_EXPORT_LOG  | Yes    | Yes    | No     | No     |
| SA_TOTAL_USAGE | Yes    | No     | No     | No     |

**I/O Specification****Input File Layout**

| Record Name | Field Name                  | Field Type | Default Value     | Description                                                                       |
|-------------|-----------------------------|------------|-------------------|-----------------------------------------------------------------------------------|
| FHEAD       | File Type Record Descriptor | Char(5)    | FHEAD             | Identifies file record type (the beginning of the input file).                    |
|             | File Line Identifier        | Number(10) | Sequential number | ID of current line being read from input file.                                    |
|             | File head descriptor        | Char(4)    | IMPA              | Describes file line type                                                          |
|             | Current date                | Char(14)   |                   | File date in YYYYMMDDHH24MISS format                                              |
| FDETL       | File Type Record Descriptor | Char(5)    | FDETL             | Identifies file record type to upload a new deal header.                          |
|             | File Line Identifier        | Number(10) | Sequential number | ID of current line being read from input file.                                    |
|             | Data source                 | Char(6)    |                   | Name of the external system that produced the file                                |
|             | New value sign              | Char(1)    |                   | Sign(+/-) for the new value.                                                      |
|             | New Value                   | Number(20) |                   | Value for the total entered by Headquarters user*10000 (4 implied decimal places) |

| Record Name | Field Name                  | Field Type | Default Value     | Description                                                                                                      |
|-------------|-----------------------------|------------|-------------------|------------------------------------------------------------------------------------------------------------------|
|             | Total seq no                | Number(20) |                   | Identifies the unique result set for this total id, total revision, store/day, Balancing group and index values. |
|             | Store                       | Number(10) |                   | Store number for a store/day combination                                                                         |
|             | Business Date               | Char(8)    |                   | Date for store/day combination                                                                                   |
|             | Total id                    | Char(10)   |                   | ID to uniquely identify the total.                                                                               |
|             | Ref no 1                    | Char(30)   |                   | The first reference value based by which the total is grouped                                                    |
|             | Ref no 2                    | Char(30)   |                   | The second reference value based by which the total is grouped                                                   |
|             | Ref no 3                    | Char(30)   |                   | The third reference value based by which the total is grouped                                                    |
| FTAIL       | File Type record descriptor | Char(5)    | FTAIL             | Identifies file record type (the end of the input file).                                                         |
|             | File Line Identifier        | Number(10) | Sequential number | ID of current line being read from input file.                                                                   |
|             | File Record Counter         | Number(10) | Sequential number | Number of records/transactions in current file (only records between head and tail)                              |

## saimptlog (Sales Audit Import)

### Functional Area

Oracle Retail Sales Audit (ReSA)

### Module Affected

SAIMPTLOG.C

SAIMPTLOGI.C

### Design Overview



Importing POS data to ReSA is a four or five-step process depending on whether saimptlogi or saimptlog is used. Saimptlog produces SQL\*Loader files while saimptlogi does inserts directly into the database. Saimptlogi is meant for use in a trickle feed environment.

To import POS data, perform the following:

1. SAGETREF must be run to generate the current reference files:

- Items
- Wastage
- Sub-transaction level items
- Primary variant relationships
- Variable weight PLU
- Store business day
- Code types
- Error codes
- Credit card validation
- Store POS
- Tender type
- Merchant code types
- Partner vendors
- Supplier vendors
- Employee ids
- Banner ids

These files are all used as input to SAIMPTLOG and SAIMPTLOGI. Because SAIMPTLOG and SAIMPTLOGI can be threaded, this boosts performance by limiting interaction with the database.

2. Either SAIMPTLOG or SAIMPTLOGI must be run against each POS file. The POS files are the transaction log files in Oracle Retail compatible format called RTLOG. The retailer is responsible for converting its transaction logs to RTLOGs. Both SAIMPTLOG and SAIMPTLOGI create a write lock for a store/day combination on ReSA tables and then set the data\_status to loading until SAIMPTLOGFIN is executed. SAIMPTLOG generates distinct SQL\*Loader files for that store/day for the sa\_tran\_head, sa\_tran\_item, sa\_tran\_disc, sa\_tran\_tax, sa\_tran\_tender, sa\_error, sa\_customer, sa\_cust\_attrib and sa\_missing\_tran tables, whereas SAIMPTLOGI inserts data to the database directly. Both produce an Oracle Retail formatted voucher file for processing.
3. SQL\*Loader is executed to load the transaction tables from the files created by SAIMPTLOG. The store/day SQL\*Loader files can be concatenated into a single file per table to optimize load times. Alternatively, multiple SQL\*Loader files can be used as input to SQL\*Loader. SQL\*Loader may not be run in parallel with itself when loading a table. Header data (primary keys) must be loaded before ancillary data (foreign keys). This means that the sa\_tran\_head table must be loaded first; sa\_tran\_item before sa\_tran\_disc; and sa\_customer before sa\_cust\_attrib. The remaining tables may be loaded in parallel.
4. SAVOUCH is executed to load each of the voucher files in Oracle Retail standard formatted. SAVOUCH may not be multiply threaded.

5. SAIMPTLOGFIN is executed to populate the sa\_balance\_group table, cancel post voided transactions and vouchers, validate missing transactions, and to mark the import as either partially or fully complete loaded. SAIMPTLOGFIN may not be multi-threaded.

---

**Note:** This design covers only step 2 and 3.

---

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                                                                  |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Sales Audit – Runs daily                                                                                                                                                                                                     |
| Scheduling Considerations | SAIMPTLOG and SAIMPTLOGI should run after the SAGETREF.PC to get the reference files as input files. RTLOGs must also be ready as input files                                                                                |
| Pre-Processing            | Saprepost saimptlog pre – change constraints on ReSA tables<br>OR<br>Saprepost saimptlogi pre – change constraints on ReSA tables.                                                                                           |
| Post-Processing           | Saprepost saimptlog post – change back constraints on ReSA tables<br>OR<br>Saprepost saimptlogi post – change back constraints on ReSA tables<br>Sqlldr – use sql loader to load data into ReSA tables (for SAIMPTLOG only). |
| Threading Scheme          | SAIMPTLOG and SAIMPTLOGI may be threaded as long as the parallel executions do not include the same store/day.                                                                                                               |

### Restart/Recovery

N/A

### Locking Strategy

This program is attempted to obtain a write lock on the store/day with a call to get\_lock library function.

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| SA_ROUNDING_RULE_HEAD   | Yes    | No     | No     | No     |
| SA_ROUNDING_RULE_DETAIL | Yes    | No     | No     | No     |
| SA_STORE_DAY            | Yes    | No     | Yes    | No     |
| SA_TRAN_HEAD            | No     | Yes    | No     | No     |
| SA_CUSTOMER             | No     | Yes    | No     | No     |
| SA_CUST_ATTRIB          | No     | Yes    | No     | No     |
| SA_TRAN_ITEM            | No     | Yes    | No     | No     |
| SA_TRAN_DISC            | No     | Yes    | No     | No     |
| SA_TRAN_TAX             | No     | Yes    | No     | No     |
| SA_TRAN_TENDER          | No     | Yes    | No     | No     |
| SA_ERROR                | No     | Yes    | No     | No     |
| SA_MISSING_TRAN         | No     | Yes    | No     | No     |
| ALL_SEQUENCES           | Yes    | No     | No     | No     |

**Shared Modules**

N/A

**I/O Specification****Input Files**

The input files for this program are reference files generated by SAGETREF.PC and RTLOGs. The file layouts of reference files are listed below. **Please refer to a separate document called ‘ReSA Interface File Layout [rtlog]’ for a detailed description of RLOGs.**

**File Name: Item File**

| Record Name | Field Name      | Field Type | Default Value | Description              |
|-------------|-----------------|------------|---------------|--------------------------|
|             | Item            | Char(25)   |               | Item Number              |
|             | Dept            | Number(4)  |               | Department Number        |
|             | Class           | Number(4)  |               | Class                    |
|             | Subclass        | Number(4)  |               | Subclass                 |
|             | Standard UOM    | Char(4)    |               | Standard Unit of Measure |
|             | Catchweight Ind | Char(1)    |               | Catchweight Indicator    |

**File Name: Waste File**

| Record Name | Field Name | Field Type | Default Value | Description |
|-------------|------------|------------|---------------|-------------|
|             | Item       | Char(25)   |               | Item Number |
|             | Waste Type | Char(6)    |               | Waste type  |
|             | Waste PCT  | Number(12) |               | Waste PCT   |

**File Name: Ref Item File**

| Record Name | Field Name | Field Type | Default Value | Description    |
|-------------|------------|------------|---------------|----------------|
|             | Ref Item   | Char(25)   |               | Reference Item |
|             | Item       | Char(25)   |               | Item Number    |

**File Name: Prim Variant File**

| Record Name | Field Name   | Field Type | Default Value | Description  |
|-------------|--------------|------------|---------------|--------------|
|             | Location     | Number(10) |               | Location Id  |
|             | Item         | Char(25)   |               | Item Number  |
|             | Prim Variant | Char(25)   |               | Prim Variant |

**File Name: Variable UPC File**

| Record Name | Field Name       | Field Type | Default Value | Description        |
|-------------|------------------|------------|---------------|--------------------|
|             | Format Id        | Char(1)    |               | Format Id          |
|             | Format Desc      | Char(20)   |               | Format Description |
|             | Prefix Lenth     | Number(1)  |               | Prefix Length      |
|             | Begin Item Digit | Number(2)  |               | Begin Item digit   |
|             | Begin VAR digit  | Number(2)  |               | Begin VAR digit    |
|             | Check digit      | Number(2)  |               | Check digit        |
|             | Default prefix   | Number(1)  |               | Default prefix     |
|             | Prefix           | Number(1)  |               | Prefix             |

**File Name: Store/Day File**

| Record Name | Field Name       | Field Type | Default Value | Description                      |
|-------------|------------------|------------|---------------|----------------------------------|
|             | Store            | Number(10) |               | Store Id                         |
|             | Business Date    | Char(8)    |               | Business date in YYYYMMDD format |
|             | Store day seq no | Number(20) |               | Store Day sequence no.           |

| Record Name | Field Name         | Field Type | Default Value | Description                     |
|-------------|--------------------|------------|---------------|---------------------------------|
|             | Day                | Number(3)  |               | Day                             |
|             | Tran No. Generated | Char(6)    |               | Tran Number generated           |
|             | POS data expected  | Char(1)    |               | Indicator for POS data expected |
|             | Currency RTL dec   | Number(1)  |               | Currency RTL dec                |
|             | Currency code      | Char(3)    |               | Currency code                   |
|             | Country Id         | Char(3)    |               | Country Id                      |

**File Name: Promotion Data File**

| Record Name | Field Name | Field Type | Default Value | Description  |
|-------------|------------|------------|---------------|--------------|
|             | Promotion  | Number(10) |               | Promotion Id |
|             | Component  | Number(10) |               | Component Id |

**File Name: Code Type data File**

| Record Name | Field Name    | Field Type | Default Value | Description   |
|-------------|---------------|------------|---------------|---------------|
|             | Code Type     | Char(4)    |               | Code Type     |
|             | Code          | Char(6)    |               | Code number   |
|             | Code Sequence | Number(4)  |               | Code sequence |

**File Name: Store POS File**

| Record Name | Field Name     | Field Type | Default Value | Description              |
|-------------|----------------|------------|---------------|--------------------------|
|             | Store          | Number(10) |               | Store Id                 |
|             | POS Type       | Char(6)    |               | Point Of Sale type       |
|             | Start Tran No. | Number(10) |               | Start Transaction Number |
|             | End Tran No.   | Number(10) |               | End Transaction Number   |

**File Name: Tender Type File**

| Record Name | Field Name        | Field Type | Default Value | Description             |
|-------------|-------------------|------------|---------------|-------------------------|
|             | Tender type group | Char(6)    |               | Tender type group       |
|             | Tender type id    | Number(6)  |               | Tender Type Id          |
|             | Tender type desc  | Char(120)  |               | Tender Type description |

**File Name: Error File**

| Record Name | Field Name   | Field Type | Default Value | Description       |
|-------------|--------------|------------|---------------|-------------------|
|             | Error Code   | Char(25)   |               | Error Code        |
|             |              |            |               |                   |
|             | Error Desc   | Char(255)  |               | Error Description |
|             | Rec solution | Char(255)  |               | Rectify solution  |

**File Name: Credit Card Validation File**

| Record Name | Field Name  | Field Type | Default Value | Description        |
|-------------|-------------|------------|---------------|--------------------|
|             | Card Length | Number(2)  |               | Credit card Length |
|             | From Prefix | Number(6)  |               | From Prefix        |
|             | To Prefix   | Number(6)  |               | To Prefix          |
|             | Card Type   | Number(6)  |               | Credit card type   |
|             | VAL Type    | Char(6)    |               | VAL Type           |

**File Name: Merchant Code Data File**

| Record Name | Field Name     | Field Type | Default Value | Description       |
|-------------|----------------|------------|---------------|-------------------|
|             | Non-Merch Code | Char(6)    |               | Non-Merchant Code |

**File Name: Partner File**

| Record Name | Field Name   | Field Type | Default Value | Description  |
|-------------|--------------|------------|---------------|--------------|
|             | Partner Type | Char(6)    |               | Partner Type |
|             | Partner Id   | Char(10)   |               | Partner Id   |

**File Name: Supplier File**

| Record Name | Field Name      | Field Type | Default Value | Description     |
|-------------|-----------------|------------|---------------|-----------------|
|             | Supplier        | Number(10) |               | Supplier Id     |
|             | Supplier Status | Char(1)    |               | Supplier status |

**File Name: Employee File**

| Record Name | Field Name | Field Type | Default Value | Description  |
|-------------|------------|------------|---------------|--------------|
|             | Store      | Number(10) |               | Store Number |
|             | POS Id     | Char(10)   |               | POS Id       |

| Record Name | Field Name  | Field Type | Default Value | Description |
|-------------|-------------|------------|---------------|-------------|
|             | Employee Id | Char(10)   |               | Employee Id |

**File Name: Banner File**

| Record Name | Field Name | Field Type | Default Value | Description  |
|-------------|------------|------------|---------------|--------------|
|             | Store      | Number(10) |               | Store Number |
|             | Banner Id  | Number(4)  |               | Banner Id    |

**Output File Layout****File Name: Sales Audit Voucher File**

| Record Name | Field Name                  | Field Type | Default Value | Description                                        |
|-------------|-----------------------------|------------|---------------|----------------------------------------------------|
| FHEAD       | File Type Record Descriptor | Char(5)    | FHEAD         | File type Record Descriptor                        |
|             | SA File Line No             | Char(10)   |               | Sales Audit File Line No.                          |
|             | Translator Id               | Char(5)    | SAVO          | Identifies transaction type                        |
|             | Sys Date                    | Char(14)   |               | System date in YYYYMMDDHHMMSS format               |
|             | Is business date            | Char(8)    |               | Business date in YYYYMMDD format                   |
| FDETL       | Record Descriptor           | Char(5)    | FDETL         | File Type Record descriptor                        |
|             | SA File Line No             | Number(10) |               | Sales Audit File Line no.                          |
|             | Voucher seq Number          | Number(20) |               | Unique identifier for an entry to sa_voucher table |
|             | Voucher No                  | Char(16)   |               | Voucher No.                                        |
|             | Voucher Type                | Number(6)  |               | Voucher Type                                       |
|             | Assigned Business Date      | Char(8)    |               | Business date in YYYYMMDD format                   |
|             | Assigned Store              | Number(10) |               | Store to which the voucher is assigned             |
|             | Issuing Date                | Char(8)    |               | Date this document was issued                      |
|             | Issuing store               | Number(10) |               | Store this document was issued from                |

| Record Name | Field Name                  | Field Type | Default Value | Description                                                                            |
|-------------|-----------------------------|------------|---------------|----------------------------------------------------------------------------------------|
|             | Issuing POS Register        | Char(5)    |               | Issuing Point Of Sale Register                                                         |
|             | Issuing Cashier             | Char(10)   |               | Issuing Cashier                                                                        |
|             | Issued Tran Seq No.         | Number(20) |               | Transaction sequence no.                                                               |
|             | Issued item seq number      | Number(4)  |               | Holds the item sequence of the item when the voucher is sold as an item (gift voucher) |
|             | Issued Tender Seq No.       | Number(4)  |               | Tender sequence no.                                                                    |
|             | Issued Amount               | Number(20) |               | Issued Amount * 10000 (4 implied digits)                                               |
|             | Issued Cust Name            | Char(120)  |               | Issued Customer Name                                                                   |
|             | Issued Customer Addr1       | Char(240)  |               | Issued Customer Addr1                                                                  |
|             | Issued Customer Addr2       | Char(240)  |               | Issued Customer Addr 2                                                                 |
|             | Issued Customer City        | Char(120)  |               | City of the customer, the voucher is issued                                            |
|             | Issued Customer State       | Char(3)    |               | State of the customer                                                                  |
|             | Issued Customer Postal Code | Char(30)   |               | Postal address of the customer.                                                        |
|             | Issued Customer Country     | Char(3)    |               | Country of the customer the voucher was issued.                                        |
|             | Recipient Name              | Char(120)  |               | Name of the intended recipient                                                         |
|             | Recipient State             | Char(3)    |               | The state of the intended recipient.                                                   |
|             | Recipient Country           | Char(3)    |               | The country of the intended recipient.                                                 |
|             | Redemption Date             | Char(8)    |               | Date the voucher was redeemed.                                                         |
|             | Redemption Store            | Number(10) |               | Store, the voucher was redeemed at.                                                    |
|             | Redemption Register         | Char(5)    |               | Register, the document was redeemed at.                                                |



| Record Name | Field Name                   | Field Type | Default Value | Description                                                                                                     |
|-------------|------------------------------|------------|---------------|-----------------------------------------------------------------------------------------------------------------|
|             | Redemption cashier           | Char(10)   |               | Cashier redeeming the voucher                                                                                   |
|             | Redemption tran seq number   | Number(20) |               | Transaction Number when the document was redeemed                                                               |
|             | Redemption Tender seq number | Number(4)  |               | This column holds the tender sequence of the tender within the transaction when a voucher is redeemed as tender |
|             | Redemption Amount            | Number(20) |               | Amount the document was redeemed for*10000 (4 implied decimal places)                                           |
|             | Expiry Date                  | Char(8)    |               | Expiry Date                                                                                                     |
|             | Status                       | Char(1)    |               | Indicator showing the document's status, issued or redeemed. Valid values = I – Issued, R - Redeemed.           |
|             | Comments                     | Char(2000) |               | Comments                                                                                                        |
| FTAIL       | Record Descriptor            | Char(5)    | FTAIL         | File Type Record descriptor                                                                                     |
|             | SA File Line No.             | Number(10) |               | Sales Audit File Line No.                                                                                       |
|             | #lines                       | Number(10) |               | Total number of transaction lines in file (not including FHEAD and FTAIL)                                       |

### Control Files

File Name: Sadisc.ctl

| Table Name   | Column Name     | Field Type       | Field Width | Position | Description |
|--------------|-----------------|------------------|-------------|----------|-------------|
| SA_TRAN_DISC | TRAN_SEQ_NO     | INTEGER EXTERNAL | 20          | 1:20     |             |
|              | ITEM_SEQ_NO     | INTEGER EXTERNAL | 4           | 21:24    |             |
|              | DISCOUNT_SEQ_NO | INTEGER EXTERNAL | 4           | 25:28    |             |

| Table Name | Column Name               | Field Type          | Field Width | Position | Description |
|------------|---------------------------|---------------------|-------------|----------|-------------|
|            | RS_PROMO_TYPE             | CHAR                | 6           | 29:34    |             |
|            | PROMOTION                 | INTEGER<br>EXTERNAL | 10          | 35:44    |             |
|            | DISC_TYPE                 | CHAR                | 6           | 45:50    |             |
|            | COUPON_NO                 | CHAR                | 16          | 51:66    |             |
|            | COUPON_REF_NO             | CHAR                | 16          | 67:82    |             |
|            | QTY                       | DECIMAL<br>EXTERNAL | 14          | 83:96    |             |
|            | UNIT_DISCOUNT<br>_AMT     | DECIMAL<br>EXTERNAL | 21          | 97:117   |             |
|            | STANDARD_QTY              | DECIMAL<br>EXTERNAL | 14          | 118:131  |             |
|            | STANDARD_UNIT<br>_DISCAMT | DECIMAL<br>EXTERNAL | 21          | 132:152  |             |
|            | REF_NO13                  | CHAR                | 30          | 153:182  |             |
|            | REF_NO14                  | CHAR                | 30          | 183:212  |             |
|            | REF_NO15                  | CHAR                | 30          | 213:242  |             |
|            | REF_NO16                  | CHAR                | 30          | 243:272  |             |
|            | ERROR_IND                 | CHAR                | 1           | 273:273  |             |
|            | CATCHWEIGHT_I<br>ND       | CHAR                | 1           | 274:274  |             |
|            | UOM_QUANTITY              | INTEGER<br>EXTERNAL | 12          | 275:286  |             |
|            | PROMO_COMP                | INTEGER<br>EXTERNAL | 10          | 287:296  |             |
|            | STORE                     | INTEGER<br>EXTERNAL | 10          | 297:306  |             |
|            | DAY                       | INTEGER<br>EXTERNAL | 3           | 307:309  |             |

File Name: Sacust.ctl

| Table Name      | Column Name | Field Type                  | Field Width | Position | Description |
|-----------------|-------------|-----------------------------|-------------|----------|-------------|
| SA_CUSTO<br>MER | TRAN_SEQ_NO | INTEGER<br>EXTERNAL<br>DATE | 20          | 1 :20    |             |

| Table Name | Column Name  | Field Type | Field Width | Position | Description             |
|------------|--------------|------------|-------------|----------|-------------------------|
|            | CUST_ID      | CHAR       | 16          | 21 :36   |                         |
|            | CUST_ID_TYPE | CHAR       | 6           | 37 :42   |                         |
|            | NAME         | CHAR       | 120         | 43 :162  |                         |
|            | ADDR1        | CHAR       | 120         | 163:282  |                         |
|            | ADDR2        | CHAR       | 120         | 283:402  |                         |
|            | CITY         | CHAR       | 120         | 403:522  |                         |
|            | STATE        | CHAR       | 3           | 523:525  |                         |
|            | POSTAL_CODE  | CHAR       | 30          | 526:555  |                         |
|            | COUNTRY      | CHAR       | 3           | 556:558  |                         |
|            | HOME_PHONE   | CHAR       | 20          | 559:578  |                         |
|            | WORK_PHONE   | CHAR       | 20          | 579:598  |                         |
|            | E_MAIL       | CHAR       | 100         | 599:698  |                         |
|            | BIRTHDATE    | DATE       | 8           | 699:706  | FORMAT IS<br>"YYYYMMDD" |

File Name: Sathead.ctl

| Table Name   | Column Name      | Field Type          | Field Width | Position | Description                           |
|--------------|------------------|---------------------|-------------|----------|---------------------------------------|
| SA_TRAN_HEAD | TRAN_SEQ_NO      | INTEGER<br>EXTERNAL | 20          | 1:20     |                                       |
|              | REV_NO           | INTEGER<br>EXTERNAL | 3           | 21:23    |                                       |
|              | STORE_DAY_SEQ_NO | INTEGER<br>EXTERNAL | 20          | 24:43    |                                       |
|              | TRAN_DATETIME    | DATE                | 14          | 44:57    | FORMAT IS<br>YYYYMM<br>DDHH24MI<br>SS |
|              | REGISTER         | CHAR                | 5           | 58:62    |                                       |
|              | TRAN_NO          | INTEGER<br>EXTERNAL | 10          | 63:72    |                                       |
|              | CASHIER          | CHAR                | 10          | 73:82    |                                       |
|              | SALESPERSON      | CHAR                | 10          | 83:92    |                                       |
|              | TRAN_TYPE        | CHAR                | 6           | 93:98    |                                       |

| Table Name | Column Name          | Field Type          | Field Width | Position | Description                                             |
|------------|----------------------|---------------------|-------------|----------|---------------------------------------------------------|
|            | SUB_TRAN_TYPE        | CHAR                | 6           | 99:104   |                                                         |
|            | ORIG_TRAN_NO         | INTEGER<br>EXTERNAL | 10          | 105:114  |                                                         |
|            | ORIG_REG_NO          | CHAR                | 5           | 115:119  |                                                         |
|            | REF_NO1              | CHAR                | 30          | 120:149  |                                                         |
|            | REF_NO2              | CHAR                | 30          | 150:179  |                                                         |
|            | REF_NO3              | CHAR                | 30          | 180:209  |                                                         |
|            | REF_NO4              | CHAR                | 30          | 210:239  |                                                         |
|            | REASON_CODE          | CHAR                | 6           | 240:245  |                                                         |
|            | VENDOR_NO            | CHAR                | 10          | 246:255  |                                                         |
|            | VENDOR_INVC_NO       | CHAR                | 30          | 256:285  |                                                         |
|            | PAYMENT_REF_NO       | CHAR                | 16          | 286:301  |                                                         |
|            | PROOF_OF_DELIVERY_NO | CHAR                | 30          | 302:331  |                                                         |
|            | STATUS               | CHAR                | 6           | 332:337  |                                                         |
|            | VALUE                | CHAR                | 22          | 338:359  | INCLUDES AN OPTIONAL NEGATIVE SIGN AND A DECIMAL POINT. |
|            | POS_TRAN_IND         | CHAR                | 1           | 360:360  |                                                         |
|            | UPDATE_ID            | CHAR                | 30          | 361:390  |                                                         |
|            | UPDATE_DATE TIME     | DATE                | 14          | 391:404  | FORMAT IS YYYYMM DDHH24MISS                             |
|            | ERROR_IND            | CHAR                | 1           | 405:405  |                                                         |
|            | BANNER_NO            | INTEGER<br>EXTERNAL | 4           | 406:409  |                                                         |
|            | CUST_ORDER_NO        | CHAR                | 30          | 410:439  |                                                         |

| Table Name | Column Name     | Field Type | Field Width | Position | Description                           |
|------------|-----------------|------------|-------------|----------|---------------------------------------|
|            | CUST_ORDER_DATE | DATE       | 14          | 440:453  | FORMAT IS<br>YYYYMM<br>DDHH24MI<br>SS |

**File Name: Satitem.ctl**

| Table Name   | Column Name     | Field Type          | Field Width | Position | Description                                                               |
|--------------|-----------------|---------------------|-------------|----------|---------------------------------------------------------------------------|
| SA_TRAN_ITEM | TRAN_SEQ_NO     | INTEGER<br>EXTERNAL | 20          | 1:20     |                                                                           |
|              | ITEM_SEQ_NO     | INTEGER<br>EXTERNAL | 4           | 21:24    |                                                                           |
|              | ITEM_STATUS     | CHAR                | 6           | 25:30    |                                                                           |
|              | ITEM_TYPE       | CHAR                | 6           | 31:36    |                                                                           |
|              | ITEM            | CHAR                | 25          | 37:61    |                                                                           |
|              | REF_ITEM        | CHAR                | 25          | 62:86    |                                                                           |
|              | NON_MERCH_ITEM  | CHAR                | 25          | 87:111   |                                                                           |
|              | VOUCHER_NO      | CHAR                | 16          | 112:127  |                                                                           |
|              | DEPT            | INTEGER<br>EXTERNAL | 4           | 128:131  |                                                                           |
|              | CLASS           | INTEGER<br>EXTERNAL | 4           | 132:135  |                                                                           |
|              | SUBCLASS        | INTEGER<br>EXTERNAL | 4           | 136:139  |                                                                           |
|              | QTY             | DECIMAL<br>EXTERNAL | 14          | 140:153  | INCLUDES<br>AN<br>OPTIONAL<br>NEGATIVE<br>SIGN AND A<br>DECIMAL<br>POINT. |
|              | UNIT_RETAIL     | DECIMAL<br>EXTERNAL | 21          | 154:174  | INCLUDES A<br>DECIMAL<br>POINT.                                           |
|              | SELLING UOM     | CHAR                | 4           | 175:178  |                                                                           |
|              | OVERRIDE_REASON | CHAR                | 6           | 179:184  |                                                                           |

| Table Name | Column Name                   | Field Type          | Field Width | Position | Description                                                                  |
|------------|-------------------------------|---------------------|-------------|----------|------------------------------------------------------------------------------|
|            | ORIG_UNIT_RETAIL              | DECIMAL<br>EXTERNAL | 21          | 185:205  | INCLUDES A<br>DECIMAL<br>POINT.                                              |
|            | STANDARD_ORIG_UN<br>IT_RETAIL | DECIMAL<br>EXTERNAL | 21          | 206:226  |                                                                              |
|            | TAX_IND                       | CHAR                | 1           | 227:227  |                                                                              |
|            | ITEM_SWIPED_IND               | CHAR                | 1           | 228:228  |                                                                              |
|            | ERROR_IND                     | CHAR                | 1           | 229:229  |                                                                              |
|            | DROP_SHIP_IND                 | CHAR                | 1           | 230:230  |                                                                              |
|            | WASTE_TYPE                    | CHAR                | 6           | 231:236  |                                                                              |
|            | WASTE_PCT                     | DECIMAL<br>EXTERNAL | 12          | 237:248  | INCLUDES A<br>DECIMAL<br>POINT.                                              |
|            | PUMP                          | CHAR                | 8           | 249:256  |                                                                              |
|            | RETURN_REASON_CO<br>DE        | CHAR                | 6           | 257:262  |                                                                              |
|            | SALESPERSON                   | CHAR                | 10          | 263:272  |                                                                              |
|            | EXPIRATION_DATE               | DATE                | 8           | 273:280  | FORMAT IS<br>YYYYMM<br>DD                                                    |
|            | STANDARD_QTY                  | DECIMAL<br>EXTERNAL | 14          | 281:294  | INCLUDES<br>AN<br>OPTIONAL<br>NEGATIVE<br>SIGN<br>AND A<br>DECIMAL<br>POINT. |
|            | STANDARD_UNIT_RE<br>TAIL      | DECIMAL<br>EXTERNAL | 21          | 295:315  | INCLUDES A<br>DECIMAL<br>POINT.                                              |
|            | STANDARD_UOM                  | CHAR                | 4           | 316:319  |                                                                              |
|            | REF_NO5                       | CHAR                | 30          | 320:349  |                                                                              |
|            | REF_NO6                       | CHAR                | 30          | 350:379  |                                                                              |
|            | REF_NO7                       | CHAR                | 30          | 380:409  |                                                                              |
|            | REF_NO8                       | CHAR                | 30          | 410:439  |                                                                              |

| Table Name | Column Name            | Field Type          | Field Width | Position | Description |
|------------|------------------------|---------------------|-------------|----------|-------------|
|            | CATCHWEIGHT_IND        | CHAR                | 1           | 440:440  |             |
|            | SELLING_ITEM           | CHAR                | 25          | 441:465  |             |
|            | CUSTOMER_ORDER_LINE_NO | INTEGER<br>EXTERNAL | 6           | 466:471  |             |
|            | MEDIA_ID               | INTEGER<br>EXTERNAL | 10          | 472:481  |             |
|            | UOM_QUANTITY           | INTEGER<br>EXTERNAL | 12          | 482:493  |             |

**File Name: Sattend.ctl**

| Table Name     | Column Name       | Field Type          | Field Width | Position | Description                                             |
|----------------|-------------------|---------------------|-------------|----------|---------------------------------------------------------|
| SA_TRAN_TENDER | TRAN_SEQ_NO       | INTEGER<br>EXTERNAL | 20          | 1:20     |                                                         |
|                | TENDER_SEQ_NO     | INTEGER<br>EXTERNAL | 4           | 21:24    |                                                         |
|                | TENDER_TYPE_GROUP | CHAR                | 6           | 25:30    |                                                         |
|                | TENDER_TYPE_ID    | INTEGER<br>EXTERNAL | 6           | 31:36    |                                                         |
|                | TENDER_AMT        | DECIMAL<br>EXTERNAL | 22          | 37:58    | INCLUDES AN OPTIONAL NEGATIVE SIGN AND A DECIMAL POINT. |
|                | CC_NO             | INTEGER<br>EXTERNAL | 40          | 59:98    |                                                         |
|                | CC_CC_EXP_DATE    | DATE                | 8           | 99:106   | FORMAT IS YYYYMMDD                                      |
|                | CC_AUTH_NO        | CHAR                | 16          | 107:122  |                                                         |
|                | CC_AUTH_SRC       | CHAR                | 6           | 123:128  |                                                         |
|                | CC_ENTRY_MODE     | CHAR                | 6           | 129:134  |                                                         |

| Table Name | Column Name        | Field Type          | Field Width | Position | Description |
|------------|--------------------|---------------------|-------------|----------|-------------|
|            | CC_CARDHOLDER_VERF | CHAR                | 6           | 135:140  |             |
|            | CC_TERM_ID         | CHAR                | 5           | 141:145  |             |
|            | CC_SPEC_COND       | CHAR                | 6           | 146:151  |             |
|            | VOUCHER_NO         | CHAR                | 16          | 152:167  |             |
|            | COUPON_NO          | CHAR                | 16          | 168:183  |             |
|            | COUPON_REF_NO      | CHAR                | 16          | 184:199  |             |
|            | REF_NO9            | CHAR                | 30          | 200:229  |             |
|            | REF_NO10           | CHAR                | 30          | 230:259  |             |
|            | REF_NO11           | CHAR                | 30          | 260:289  |             |
|            | REF_NO12           | CHAR                | 30          | 290:319  |             |
|            | ERROR_IND          | CHAR                | 1           | 320:320  |             |
|            | STORE              | INTEGER<br>EXTERNAL | 10          | 321:330  |             |
|            | DAY                | INTEGER<br>EXTERNAL | 3           | 331:333  |             |

File Name: Samisstr.ctl

| Table Name      | Column Name      | Field Type          | Field Width | Position | Description |
|-----------------|------------------|---------------------|-------------|----------|-------------|
| SA_MISSING_TRAN | MISS_TRAN_SEQ_NO | INTEGER<br>EXTERNAL | 20          | 1:20     |             |
|                 | STORE_DAY_SEQ_NO | INTEGER<br>EXTERNAL | 20          | 21:40    |             |
|                 | REGISTER         | CHAR                | 5           | 41:45    |             |
|                 | TRAN_NO          | INTEGER<br>EXTERNAL | 10          | 46:55    |             |
|                 | STATUS           | CHAR                | 6           | 56:61    |             |

File Name: Sattax.ctl



| Table Name      | Column Name | Field Type          | Field Width | Position | Description                                                              |
|-----------------|-------------|---------------------|-------------|----------|--------------------------------------------------------------------------|
| SA_TRAN_T<br>AX | TRAN_SEQ_NO | INTEGER<br>EXTERNAL | 20          | 1:20     |                                                                          |
|                 | TAX_CODE    | CHAR                | 6           | 21:26    |                                                                          |
|                 | TAX_SEQ_NO  | INTEGER<br>EXTERNAL | 4           | 27:30    |                                                                          |
|                 | TAX_AMT     | DECIMAL<br>EXTERNAL | 22          | 31:52    | INCLUDES<br>AN<br>OPTIONAL<br>NEGATIVE<br>SIGN AND A<br>DECIMAL<br>POINT |
|                 | ERROR_IND   | CHAR                | 1           | 53:53    |                                                                          |
|                 | REF_NO17    | CHAR                | 30          | 54:83    |                                                                          |
|                 | REF_NO18    | CHAR                | 30          | 84:113   |                                                                          |
|                 | REF_NO19    | CHAR                | 30          | 114:143  |                                                                          |
|                 | REF_NO20    | CHAR                | 30          | 144:173  |                                                                          |
|                 | STORE       | INTEGER<br>EXTERNAL | 10          | 174:183  |                                                                          |
|                 | DAY         | INTEGER<br>EXTERNAL | 3           | 184:186  |                                                                          |

File Name: Sacustatt.ctl

| Table Name         | Column Name  | Field Type          | Field Width | Position | Description |
|--------------------|--------------|---------------------|-------------|----------|-------------|
| SA_CUST_A<br>TTRIB | TRAN_SEQ_NO  | INTEGER<br>EXTERNAL | 20          | 1:20     |             |
|                    | ATTRIB_SEQSO | CHAR                | 4           | 21:24    |             |
|                    | ATTRIB_TYPE  | CHAR                | 6           | 25:30    |             |
|                    | ATTRIB_VALUE | CHAR                | 6           | 31:36    |             |

File Name: Saerror.ctl

| Table Name | Column Name         | Field Type          | Field Width | Position | Description                     |
|------------|---------------------|---------------------|-------------|----------|---------------------------------|
| SA_ERROR   | ERROR_SEQ_NO        | INTEGER<br>EXTERNAL | 20          | 1:20     |                                 |
|            | STORE_DAY_SEQ_NO    | INTEGER<br>EXTERNAL | 20          | 21:40    |                                 |
|            | BAL_GROUP_SEQ_NO    | INTEGER<br>EXTERNAL | 20          | 41:60    |                                 |
|            | TOTAL_SEQ_NO        | INTEGER<br>EXTERNAL | 20          | 61:80    |                                 |
|            | TRAN_SEQ_NO         | INTEGER<br>EXTERNAL | 20          | 81:100   |                                 |
|            | ERROR_CODE          | CHAR                | 25          | 101:125  |                                 |
|            | KEY_VALUE_1         | INTEGER<br>EXTERNAL | 4           | 126:129  |                                 |
|            | KEY_VALUE_2         | INTEGER<br>EXTERNAL | 4           | 130:133  |                                 |
|            | REC_TYPE            | CHAR                | 6           | 134:139  |                                 |
|            | STORE_OVERRIDE_IND  | CHAR                | 1           | 140:140  |                                 |
|            | HQ_OVERRIDE_IND     | CHAR                | 1           | 141:141  |                                 |
|            | UPDATE_ID           | CHAR                | 30          | 142:171  |                                 |
|            | UPDATE_DATE<br>TIME | DATE                | 14          | 172:185  | FORMAT IS<br>"YYYYMMDDHH24MISS" |
|            | ORIG_VALUE          | CHAR                | 50          | 186:235  |                                 |

## ReSA Interface File Layout [rtlog]

The following illustrates the file layout format of the Oracle Retail TLOG. The content of each Oracle Retail TLOG file is per store per day. The filename convention is RTLOG\_STORE\_DATETIME.DAT (e.g. RTLOG\_1234\_01221989010000.DAT)

FHEAD (Only 1 per file, required)  
 THEAD (Multiple expected, one per transaction, required for each transaction)  
 TCUST (Only 1 per THEAD record allowed, optional for some transaction types, see table below)  
 CATT (Attribute record specific to the TCUST record - Multiple allowed, only valid if TCUST exists)  
 TITEM (Multiple allowed per transaction, optional for some transaction types, see table below)  
 IDISC (Discount record specific to the TITEM record - Multiple allowed per item, optional see table below)  
 TTAX (Multiple allowed per transaction, optional see table below)  
 TTEND (Multiple allowed per transaction, optional for some transaction types, see table below)  
 TTAIL (1 per THEAD, required)  
 FTAIL (1 per file, required)

The order of the records within the transaction layout above is important. It aids processing by ensuring that information is present when it is needed.

| Record Name        | Field Name                  | Field Type | Default Value                | Description                                                                                           | Required | Justification/Padding |
|--------------------|-----------------------------|------------|------------------------------|-------------------------------------------------------------------------------------------------------|----------|-----------------------|
| File Header        | File Type Record Descriptor | Char(5)    | FHEAD                        | Identifies file record type                                                                           | Y        | Left/Blank            |
|                    | File Line Identifier        | Number(10) | Specified by external system | ID of current line being processed by input file.                                                     | Y        | Right/0               |
|                    | File Type Definition        | Char(4)    | RTLGL                        | Identifies file as 'Oracle Retail TLOG'.                                                              | Y        | Left/Blank            |
|                    | File Create Date            | Char(14)   | Create date                  | Date and time file was written by external system (YYYYMMDDHHMMSS).                                   | Y        | Left/None             |
|                    | Business Date               | Char(8)    | Business Date to process     | Business date of transactions. (YYYYMMDD).                                                            | Y        | Left/None             |
|                    | Location Number             | Char(10)   | Specified by external system | Store or warehouse identifier.                                                                        | Y        | Left/None             |
|                    | Reference Number            | Char(30)   | Specified by external system | This may contain the Polling ID associated with the consolidated TLOG file or used for other purpose. | N        | Left/Blank            |
| Transaction Header | File Type Record Descriptor | Char(5)    | THEAD                        | Identifies file record type.                                                                          | Y        | Left/Blank            |

| Record Name | Field Name           | Field Type | Default Value                                                                                                                                                                                       | Description                                                                                                                                              | Required | Justification/Padding |
|-------------|----------------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|----------|-----------------------|
|             | File Line Identifier | Number(10) | Specified by external system                                                                                                                                                                        | ID of current line being processed by input file.                                                                                                        | Y        | Right/0               |
|             | Register             | Char(5)    |                                                                                                                                                                                                     | Till used at store.                                                                                                                                      | Y        | Left/Blank            |
|             | Transaction Date     | Char(14)   | Transaction date                                                                                                                                                                                    | Date transactions were processed at the POS (YYYYMMDDHHMMSS).                                                                                            | Y        | Left/None             |
|             | Transaction Number   | Number(10) |                                                                                                                                                                                                     | Transaction identifier.                                                                                                                                  | Y        | Right/0               |
|             | Cashier              | Char(10)   |                                                                                                                                                                                                     | Cashier identifier.                                                                                                                                      | N        | Left/Blank            |
|             | Salesperson          | Char(10)   |                                                                                                                                                                                                     | Salesperson identifier.                                                                                                                                  | N        | Left/Blank            |
|             | Transaction Type     | Char(6)    | Refer to 'TRAT' code_type for a list of valid types.                                                                                                                                                | Transaction type.                                                                                                                                        | Y        | Left/Blank            |
|             | Sub-transaction type | Char(6)    | Refer to 'TRAS' code_type for a list of valid types.                                                                                                                                                | Sub-transaction type. For sale, it can be employee, drive-off etc.                                                                                       | N        | Left/Blank            |
|             | Orig_tran_no         | Number(10) |                                                                                                                                                                                                     | Populated only for post-void transactions. Transaction number for the original tran that is cancelled.                                                   | N        | Right/0               |
|             | Orig_reg_no          | Char(5)    |                                                                                                                                                                                                     | Populated only for post-void transactions. Register number from the original tran.                                                                       | N        | Left/Blank            |
|             | Reason Code          | Char(6)    | Refer to 'REAC' code_type for a list of valid codes. If the transaction type is 'PAIDOU' and the sub transaction type is 'MV' or 'EV' then the valid codes come from the non_merch_code_head table. | Reason entered by cashier for some transaction types. Required for Paid In and Paid out transaction types, but can also be used for voids, returns, etc. | N        | Left/Blank            |

| Record Name | Field Name               | Field Type | Default Value                                        | Description                                                                                                                                                                                       | Required              | Justification/P adding                                      |
|-------------|--------------------------|------------|------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|-------------------------------------------------------------|
|             | Vendor Number            | Char(10)   |                                                      | Supplier id for a merchandise vendor paid out transaction, partner id for an expense vendor paid out transaction.                                                                                 | N                     | Left/Blank                                                  |
|             | Vendor Invoice Number    | Char(30)   |                                                      | Invoice number for a vendor paid out transaction.                                                                                                                                                 | N                     | Left/Blank                                                  |
|             | Payment Reference Number | Char(16)   |                                                      | The reference number of the tender used for a vendor payout. This could be the money order number, check number, etc.                                                                             | N                     | Left/Blank                                                  |
|             | Proof of Delivery Number | Char(30)   |                                                      | Proof of receipt number given by the vendor at the time of delivery. This field is populated for a vendor paid out transaction.                                                                   | N                     | Left/Blank                                                  |
|             | Reference Number 1       | Char(30)   |                                                      | Number associated with a particular transaction, for example weather for a Store Conditions transaction.<br>The SA_REFERENCE table defines what this field can contain for each transaction type. | N                     | Left/Blank                                                  |
|             | Reference Number 2       | Char(30)   |                                                      | Second generic reference number.                                                                                                                                                                  | N                     | Left/Blank                                                  |
|             | Reference Number 3       | Char(30)   |                                                      | Third generic reference number.                                                                                                                                                                   | N                     | Left/Blank                                                  |
|             | Reference Number 4       | Char(30)   |                                                      | Fourth generic reference number.                                                                                                                                                                  | N                     | Left/Blank                                                  |
|             | Value Sign               | Char(1)    | Refer to 'SIGN' code_type for a list of valid codes. | Sign of the value.                                                                                                                                                                                | Y if Value is present | Left/None                                                   |
|             | Value                    | Number(20) |                                                      | Value with 4 implied decimal places. Populated by the retailer for TOTAL trans, populated by Oracle Retail sales audit for SALE, RETURN trans.                                                    | Y if tran is a TOTAL. | Right/0 when value is present. Blank when no value is sent. |

| Record Name          | Field Name                  | Field Type | Default Value                                       | Description                                       | Required                   | Justification/Padding |
|----------------------|-----------------------------|------------|-----------------------------------------------------|---------------------------------------------------|----------------------------|-----------------------|
|                      | Banner id                   | Number(4)  |                                                     | Banner ID of the location                         | Y if multichannel_ind is Y | Right/0               |
|                      | Customer order head id      | Char(30)   |                                                     | Customer head order number.                       | N                          | Left/Blank            |
|                      | Customer order head date    | Char(14)   |                                                     | Customer order date                               | N                          | Left/None             |
| Transaction Customer | File Type Record Descriptor | Char(5)    | TCUST                                               | Identifies file record type                       | Y                          | Left/Blank            |
|                      | File Line Identifier        | Number(10) | Specified by external system                        | ID of current line being processed by input file. | Y                          | Right/0               |
|                      | Customer ID                 | Char(16)   | Customer identifier                                 | The ID number of a customer.                      | Y                          | Left/Blank            |
|                      | Customer ID type            | Char(6)    | Refer to 'CIDT' code_type for a list of valid types | Customer ID type.                                 | Y                          | Left/Blank            |
|                      | Customer Name               | Char(120)  |                                                     | Customer name.                                    | N                          | Left/Blank            |
|                      | Address 1                   | Char(240)  |                                                     | Customer address.                                 | N                          | Left/Blank            |
|                      | Address 2                   | Char(240)  |                                                     | Additional field for customer address.            | N                          | Left/Blank            |
|                      | City                        | Char(120)  |                                                     | City.                                             | N                          | Left/Blank            |
|                      | State                       | Char(3)    | State identifier                                    | State.                                            | N                          | Left/Blank            |
|                      | Zip Code                    | Char(30)   | Zip identifier                                      | Zip code.                                         | N                          | Left/Blank            |
|                      | Country                     | Char(3)    |                                                     | Country.                                          | N                          | Left/Blank            |
|                      | Home Phone                  | Char(20)   |                                                     | Telephone number at home.                         | N                          | Left/Blank            |
|                      | Work Phone                  | Char(20)   |                                                     | Telephone number at work.                         | N                          | Left/Blank            |
|                      | E-mail                      | Char(100)  |                                                     | E-mail address.                                   | N                          | Left/Blank            |
|                      | Birthdate                   | Char(8)    |                                                     | Date of birth. (YYYYMMDD)                         | N                          | Left/Blank            |
| Customer Attribute   | File Type Record Descriptor | Char(5)    | CATT                                                | Identifies file record type                       | Y                          | Left/Blank            |

| Record Name      | Field Name                  | Field Type | Default Value                                                   | Description                                                                                       | Required | Justification/Padding |
|------------------|-----------------------------|------------|-----------------------------------------------------------------|---------------------------------------------------------------------------------------------------|----------|-----------------------|
|                  | File Line Identifier        | Number(10) | Specified by external system                                    | ID of current line being processed by input file.                                                 | Y        | Right/0               |
|                  | Attribute type              | Char(6)    | Refer to 'SACA' code_type for a list of valid types             | Type of customer attribute                                                                        | Y        | Left/Blank            |
|                  | Attribute value             | Char(6)    | Refer to members of 'SACA' code_type for a list of valid values | Value of customer attribute.                                                                      | Y        | Left/Blank            |
| Transaction Item | File Type Record Descriptor | Char(5)    | TITEM                                                           | Identifies file record type.                                                                      | Y        | Left/Blank            |
|                  | File Line Identifier        | Number(10) | Specified by external system                                    | ID of current line being processed by input file.                                                 | Y        | Right/0               |
|                  | Item Status                 | Char(6)    | Refer to 'SASI' code_type for a list of valid codes.            | Status of the item within the transaction, V for item void, S for sold item, R for returned item. | Y        | Left/Blank            |
|                  | Item Type                   | Char(6)    | Refer to 'SAIT' code_type for a list of valid codes.            | Identifies what type of item is transmitted.                                                      | Y        | Left/Blank            |
|                  | Item number type            | Char(6)    | Refer to 'UPCT' code_type for a list of valid codes.            | Identifies type of item number if item type is ITEM or REF.                                       | N        | Left/Blank            |
|                  | Format ID                   | Char(1)    | VPLU format ID.                                                 | Used to interpret VPLU items.                                                                     | N        | Left/Blank            |
|                  | Item                        | Char(25)   | Item identifier                                                 | Identifies merchandise item.                                                                      | N        | Left/Blank            |
|                  | Reference Item              | Char(25)   | Item identifier                                                 | Identifies sub-transaction level merchandise item.                                                | N        | Left/Blank            |
|                  | Non-Merchandise Item        | Char(25)   | Item identifier                                                 | Identifies non-merchandise item.                                                                  | N        | Left/Blank            |
|                  | Voucher                     | Char(16)   |                                                                 | Gift certificate number                                                                           | N        | Right/0               |

| Record Name | Field Name             | Field Type | Default Value                                        | Description                                                                                                                                                    | Required                              | Justification/P adding |
|-------------|------------------------|------------|------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------|------------------------|
|             | Department             | Number(4)  |                                                      | Identifies the department this item belongs to.<br>This is filled in by saimptlog.                                                                             | N                                     | Right/Blank            |
|             | Class                  | Number(4)  | Item's class                                         | Class of item sold or returned. Not required from a retailer, populated by Oracle Retail sales audit.<br>This is filled in by saimptlog.                       | N                                     | Right/Blank            |
|             | Subclass               | Number(4)  | Item's subclass                                      | Subclass of item sold or returned. Not required from a retailer, populated by Oracle Retail sales audit.<br>This is filled in by saimptlog.                    | N                                     | Right/Blank            |
|             | Quantity Sign          | Char(1)    | Refer to 'SIGN' code_type for a list of valid codes. | Sign of the quantity                                                                                                                                           | Y                                     | Left/None              |
|             | Quantity               | Number(12) |                                                      | Number of items purchased with 4 decimal places.                                                                                                               | Y                                     | Right/0                |
|             | SellingUnit of Measure | Char(4)    |                                                      | Unit of measure of item's quantity.                                                                                                                            | Y                                     | Left/None              |
|             | Unit Retail            | Number(20) |                                                      | Unit retail with 4 implied decimal places.                                                                                                                     | Y                                     | Right/0                |
|             | Override Reason        | Char(6)    | Refer to 'ORRC' code_type for a list of valid codes. | This column is populated when an item's price is overridden at the POS to define why it was overridden.                                                        | Y if unit retail was manually entered | Left/Blank             |
|             | Original Unit Retail   | Number(20) |                                                      | Value with 4 implied decimal places.<br>This column is populated when the item's price was overridden at the POS and the item's original unit retail is known. | Y if unit retail was manually entered | Right/0                |
|             | Taxable Indicator      | Char(1)    | Refer to 'YSNO' code_type for a list of valid codes. | Indicates whether or not item is taxable.                                                                                                                      | Y                                     | Left/None              |
|             | Pump                   | Char(8)    |                                                      | Fuel pump identifier.                                                                                                                                          | N                                     | Left/Blank             |



| Record Name | Field Name             | Field Type | Default Value                                        | Description                                                                                                                                                                               | Required | Justification/Padding |
|-------------|------------------------|------------|------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|-----------------------|
|             | Reference Number 5     | Char(30)   |                                                      | Number associated with a particular item within a transaction, for example special order number.<br>The sa_reference table defines what this field can contain for each transaction type. | N        | Left/Blank            |
|             | Reference Number 6     | Char(30)   |                                                      | Second generic reference number at the item level.                                                                                                                                        | N        | Left/Blank            |
|             | Reference Number 7     | Char(30)   |                                                      | Third generic reference number at the item level.                                                                                                                                         | N        | Left/Blank            |
|             | Reference Number 8     | Char(30)   |                                                      | Fourth generic reference number at the item level.                                                                                                                                        | N        | Left/Blank            |
|             | Item_swiped_ind        | Char(1)    | Refer to 'YSNO' code_type for a list of valid codes. | Indicates if the item was automatically entered into the POS system or if it had to be manually keyed.                                                                                    | Y        | Left/None             |
|             | Return Reason Code     | Char(6)    | Refer to 'SARR' code_type for a list of valid codes. | The reason an item was returned.                                                                                                                                                          | N        | Left/Blank            |
|             | Salesperson            | Char(10)   |                                                      | The salesperson who sold the item.                                                                                                                                                        | N        | Left/Blank            |
|             | Expiration_date        | Char(8)    |                                                      | Gift certificate expiration date (YYYYMMDD).                                                                                                                                              | N        |                       |
|             | Drop Ship Ind          | Char(1)    | Refer to 'YSNO' code type for a list of valid codes. | Indicates whether item is part of a drop shipment.                                                                                                                                        | Y        | Left/None             |
|             | Uom_qty                | Number(12) |                                                      | Qty of items purchased in the given UOM with 4 decimal places.                                                                                                                            | Y        | Right/0               |
|             | Catchweight_ind        | Char(1)    | Valid values are 'Y', 'N'                            | Identifies if the item is a catchweight item                                                                                                                                              |          | Left/None             |
|             | Selling item           | Char(25)   | Item identifier                                      | Identifies selling item.                                                                                                                                                                  | N        | Left/Blank            |
|             | Customer order line no | Number(6)  |                                                      | Identifies the customer order number                                                                                                                                                      | N        | Left/Blank            |

| Record Name   | Field Name                  | Field Type | Default Value                                        | Description                                                                                                                                          | Required    | Justification/P adding |
|---------------|-----------------------------|------------|------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|------------------------|
|               | Media id                    | Number(10) |                                                      | Identifies the customer media id                                                                                                                     | N           | Left/Blank             |
| Item Discount | File Type Record Descriptor | Char(5)    | IDISC                                                | Identifies file record type                                                                                                                          | Y           | Left/Blank             |
|               | File Line Identifier        | Number(10) | Specified by external system                         | ID of current line being processed by input file.                                                                                                    | Y           | Right/0                |
|               | RMS Promotion Number        | Char(6)    | Refer to 'PRMT' code_type for a list of valid types  | The RMS promotion type.                                                                                                                              | Y           | Left/Blank             |
|               | Discount Reference Number   | Number(10) |                                                      | Discount reference number is associated with the discount type (that is, if discount type is a promotion, this contains the promotion number).       | N           | Left/Blank             |
|               | Discount Type               | Char(6)    | Refer to 'SADT' code_type for a list of valid types. | The type of discount within a promotion. This allows a retailer to further break down coupon discounts within the "In-store" promotion, for example. | N           | Left/Blank             |
|               | Coupon Number               | Char(16)   |                                                      | Number of a store coupon used as a discount.                                                                                                         | Y if coupon | Left/Blank             |
|               | Coupon Reference Number     | Char(16)   |                                                      | Additional information about the coupon, usually contained in a second bar code on the coupon.                                                       | Y if coupon | Left/Blank             |
|               | Quantity Sign               | Char(1)    | Refer to 'SIGN' code_type for a list of valid codes. | Sign of the quantity.                                                                                                                                | Y           | Left/None              |
|               | Quantity                    | Number(12) |                                                      | The quantity purchased that discount is applied with 4 implied decimal places.                                                                       | Y           | Right/0                |
|               | Unit Discount Amount        | Number(20) |                                                      | Unit discount amount for this item with 4 implied decimal places.                                                                                    | Y           | Right/0                |

| Record Name     | Field Name                  | Field Type | Default Value                                        | Description                                                                                                                                                          | Required | Justification/Padding |
|-----------------|-----------------------------|------------|------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|-----------------------|
|                 | Reference Number 13         | Char(30)   |                                                      | Number associated with a particular transaction type at the discount level.<br>The sa_reference table defines what this field can contain for each transaction type. | N        | Left/Blank            |
|                 | Reference Number 14         | Char(30)   |                                                      | Second generic reference number at the discount level.                                                                                                               | N        | Left/Blank            |
|                 | Reference Number 15         | Char(30)   |                                                      | Third generic reference number at the discount level.                                                                                                                | N        | Left/Blank            |
|                 | Reference Number 16         | Char(30)   |                                                      | Fourth generic reference number at the discount level.                                                                                                               | N        | Left/Blank            |
|                 | Uom_qty                     | Number(12) |                                                      | Qty of items purchased in the given UOM with 4 decimal places.                                                                                                       | Y        | Right/0               |
|                 | Catchweight_ind             | Char(1)    | Valid values are 'Y', 'N'                            | Identifies if the item is a catchweight item                                                                                                                         |          | Left/None             |
|                 | Promo component             | Number(10) |                                                      | If the discount is a promotion, this field contains the promotion component value associated with the Promotion (discount reference number)                          | N        | Left/Blank            |
| Transaction Tax | File Type Record Descriptor | Char(5)    | TTAX                                                 | Identifies file record type                                                                                                                                          | Y        | Left/Blank            |
|                 | File Line Identifier        | Number(10) | Specified by external system                         | ID of current line being processed by input file.                                                                                                                    | Y        | Right/0               |
|                 | Tax Code                    | Char(6)    | Refer to 'TAXC' code_type for a list of valid codes  | Tax code to represent whether it is a state tax type, provincial tax, etc.                                                                                           | Y        | Left/Blank            |
|                 | Tax Sign                    | Char(1)    | Refer to 'SIGN' code_type for a list of valid codes. | Sign of Tax Amount.                                                                                                                                                  | Y        | Left/None             |
|                 | Tax Amount                  | Number(20) |                                                      | Amount of tax charged for this tax code type in a transaction with 4 implied decimal places.                                                                         | Y        | Right/0               |
|                 | Reference Number 17         | Char(30)   |                                                      | Generic reference number.                                                                                                                                            | N        | Left/Blank            |

| Record Name        | Field Name                  | Field Type | Default Value                                                      | Description                                                                    | Required         | Justification/Padding |
|--------------------|-----------------------------|------------|--------------------------------------------------------------------|--------------------------------------------------------------------------------|------------------|-----------------------|
|                    | Reference Number 18         | Char(30)   |                                                                    | Generic reference number.                                                      | N                | Left/Blank            |
|                    | Reference Number 19         | Char(30)   |                                                                    | Generic reference number.                                                      | N                | Left/Blank            |
|                    | Reference Number 20         | Char(30)   |                                                                    | Generic reference number.                                                      | N                | Left/Blank            |
|                    |                             |            |                                                                    |                                                                                |                  |                       |
| Transaction Tender | File Type Record Descriptor | Char(5)    | TTEND                                                              | Identifies file record type                                                    | Y                | Left/Blank            |
|                    | File Line Identifier        | Number(10) | Specified by external system                                       | ID of current line being processed by input file.                              | Y                | Right/0               |
|                    | Tender Type Group           | Char(6)    | Refer to 'TENT' code_type for as list of valid types               | High-level grouping of tender types.                                           | Y                | Left/Blank            |
|                    | Tender Type ID              | Number(6)  | Refer to the pos_tender_type_head table for as list of valid types | Low-level grouping of tender types.                                            | Y                | Left/Blank            |
|                    | Tender Sign                 | Char(1)    | Refer to 'SIGN' code_type for a list of valid codes.               | Sign of the value.                                                             | Y                | Left/None             |
|                    | Tender Amount               | Number(20) |                                                                    | Amount paid with this tender in the transaction with 4 implied decimal places. | Y                | Right/0               |
|                    | Cc_no                       | Char(40)   |                                                                    | Credit card number                                                             | Y if credit card | Left/Blank            |
|                    | Cc_auth_no                  | Char(16)   |                                                                    | Authorization number for a cc                                                  | Y if credit card | Left/Blank            |
|                    | cc authorization source     | Char(6)    | Refer to 'CCAS' code_type for as list of valid types               |                                                                                | Y if credit card | Left/Blank            |
|                    | cc cardholder verification  | Char(6)    | Refer to 'CCVF' code_type for as list of valid types               |                                                                                | Y if credit card | Left/Blank            |

| Record Name         | Field Name                  | Field Type | Default Value                                        | Description                                                                                                                                                        | Required         | Justification/Padding |
|---------------------|-----------------------------|------------|------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|-----------------------|
|                     | cc expiration date          | Char(8)    |                                                      | (YYYYMMDD)                                                                                                                                                         | Y if credit card | Left/Blank            |
|                     | cc entry mode               | Char(6)    | Refer to 'CCEM' code_type for as list of valid types | Indicates whether the credit card was swiped, thus automatically entered, or manually keyed.                                                                       | Y if credit card | Left/Blank            |
|                     | cc terminal id              | Char(5)    |                                                      | Terminal number transaction was sent from.                                                                                                                         | N                | Left/Blank            |
|                     | cc special condition        | Char(6)    | Refer to 'CCSC' code_type for as list of valid types |                                                                                                                                                                    | Y if credit card | Left/Blank            |
|                     | Voucher_no                  | Char(16)   |                                                      | Gift certificate or credit voucher serial number.                                                                                                                  | Y if voucher     | Right/0               |
|                     | Coupon Number               | Char(16)   |                                                      | Number of a manufacturer's coupon used as a tender.                                                                                                                | Y if coupon      | Left/Blank            |
|                     | Coupon Reference Number     | Char(16)   |                                                      | Additional information about the coupon, usually contained in a second bar code on the coupon.                                                                     | Y if coupon      | Left/Blank            |
|                     | Reference No 9              | Char(30)   |                                                      | Number associated with a particular transaction type at the tender level.<br>The sa_reference table defines what this field can contain for each transaction type. | N                | Left/Blank            |
|                     | Reference No 10             | Char(30)   |                                                      | Second generic reference no at the tender level.                                                                                                                   | N                | Left/Blank            |
|                     | Reference No 11             | Char(30)   |                                                      | Third generic reference no at the tender level.                                                                                                                    | N                | Left/Blank            |
|                     | Reference No 12             | Char(30)   |                                                      | Fourth generic reference no at the tender level.                                                                                                                   | N                | Left/Blank            |
| Transaction Trailer | File Type Record Descriptor | Char(5)    | TTAIL                                                | Identifies file record type                                                                                                                                        | Y                | Left/Blank            |
|                     | File Line Identifier        | Number(10) | Specified by external system                         | ID of current line being processed by input file.                                                                                                                  | Y                | Right/0               |

| Record Name  | Field Name                  | Field Type | Default Value                | Description                                                                          | Required | Justification/P adding |
|--------------|-----------------------------|------------|------------------------------|--------------------------------------------------------------------------------------|----------|------------------------|
|              | Transaction Record Counter  | Number(10) |                              | No of records processed in current tran (only records between trans head & tail)     |          |                        |
| File Trailer | File Type Record Descriptor | Char(5)    | FTAIL                        | Identifies file record type                                                          | Y        | Left/Blank             |
|              | File Line Identifier        | Number(10) | Specified by external system | ID of current line being processed by input file.                                    | Y        | Right/0                |
|              | File Record Counter         | Number(10) |                              | No of transactions processed in current file (only records between file head & tail) | Y        | Right/0                |

The RTLOG file is imported into the Sales Audit tables after validation by the batch program saimptlog. This section describes the requirements and validations performed on the records.

### Common requirements/validations

This section details the common requirements and validations performed on all transactions. The following sections describe the specific requirements of each type of transaction. If a transaction is not mentioned, then it does not have specific requirements.

#### Record Type Requirements:

| Transaction Type | Includes item records? | Includes tender records? | Includes tax records? | Includes customer records? |
|------------------|------------------------|--------------------------|-----------------------|----------------------------|
| OPEN             | No                     | No                       | No                    | No                         |
| NOSALE           | No                     | Optional                 | No                    | No                         |
| VOID             | Optional               | Optional                 | Optional              | Optional                   |
| PVOID            | No                     | No                       | No                    | No                         |
| SALE             | Yes                    | Yes                      | Optional              | Optional                   |
| RETURN           | Yes                    | Yes                      | Optional              | Optional                   |
| EEXCH            | Yes                    | No                       | Optional              | Optional                   |
| PAIDIN           | No                     | Yes                      | No                    | No                         |
| PAIDOU           | No                     | Yes                      | No                    | No                         |
| PULL             | No                     | Yes                      | No                    | No                         |
| LOAN             | No                     | Yes                      | No                    | No                         |
| COND             | No                     | No                       | No                    | No                         |
| CLOSE            | No                     | No                       | No                    | No                         |
| TOTAL            | No                     | No                       | No                    | No                         |

| Transaction Type | Includes item records?                                                                                                                                                                            | Includes tender records? | Includes tax records? | Includes customer records? |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|-----------------------|----------------------------|
| REFUND           | This transaction is not sent through the RTLOG. It is entered at the HQ level. The TITEM and TCUST records are optional. The TTEND record is required. A TTAX record should not be included.      |                          |                       |                            |
| METER            | Yes                                                                                                                                                                                               | No                       | No                    | No                         |
| PUMPT            | Yes                                                                                                                                                                                               | No                       | No                    | No                         |
| TANKDP           | Yes                                                                                                                                                                                               | No                       | No                    | No                         |
| TERM             | TERM records are created by saimptlog and then loaded into the database. They do not come from the RTLOG file. They require one TITEM, one TTEND, one TTAX, one TCUST record and one CATT record. |                          |                       |                            |
| DCLOSE           | No                                                                                                                                                                                                | No                       | No                    | No                         |

Requirements per record type:

| Record Type | Requirements                                                         |
|-------------|----------------------------------------------------------------------|
| IDISC       | IDISC records must immediately follow their associated TITEM record. |
| CATT        | CATT records must immediately follow their associated TCUST record.  |

Code Type Validations:

| Record Name        | Field Name           | Code Type                                                                                                                                                                                                                                                  |
|--------------------|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Transaction Header | Transaction Type     | TRAT                                                                                                                                                                                                                                                       |
|                    | Sub-transaction Type | TRAS                                                                                                                                                                                                                                                       |
|                    | Reason Code          | REAC or values from non_merch_code_head if the transaction type is 'PAIDOU' and the sub transaction type is 'MV' or 'EV'.                                                                                                                                  |
|                    | Value Sign           | SIGN                                                                                                                                                                                                                                                       |
|                    | Vender No            | If the transaction type is 'PAIDOU' and the sub transaction type is 'MV', this field is validated against the supplier table. If the transaction type is 'PAIDOU' and the sub transaction type is 'EV', this field is validated against the partner table. |
| Transaction Item   | Item Type            | SAIT                                                                                                                                                                                                                                                       |
|                    | Item Status          | SASI                                                                                                                                                                                                                                                       |
|                    | Item Number Type     | UPCT                                                                                                                                                                                                                                                       |
|                    | Quantity Sign        | SIGN                                                                                                                                                                                                                                                       |

| Record Name          | Field Name                 | Code Type                         |
|----------------------|----------------------------|-----------------------------------|
|                      | Taxable Indicator          | YSNO                              |
|                      | Price Override Reason Code | ORRC                              |
|                      | Item Swiped Indicator      | YSNO                              |
|                      | Return Reason Code         | SARR                              |
| Item Discount        | RMS Promotion Type         | PRMT                              |
|                      | Discount Type              | SADT                              |
|                      | Quantity Sign              | SIGN                              |
| Transaction Customer | Customer ID Type           | CIDT                              |
| Customer Attribute   | Attribute Type             | SACA                              |
|                      | Attribute value            | Code types from codes in SACA.    |
| Transaction Tax      | Tax code                   | TAXC                              |
|                      | Tax sign                   | SIGN                              |
| Transaction Tender   | Tender Type Group          | TENT                              |
|                      | Tender Sign                | SIGN                              |
|                      | Tender Type ID             | <i>Pos_tender_type_head table</i> |
|                      | CC Authorization Source    | CCAS                              |
|                      | CC Cardholder Verification | CCVF                              |
|                      | CC Entry Mode              | CCEM                              |
|                      | CC Special Condition       | CCSC                              |

Dates are validated: Business Date, Transaction Date, Expiration Date Also, saimptlog accepts only business dates that are within the PERIOD.VDATE minus the SA\_SYSTEM\_OPTIONS.DAYS\_POST\_SALE value.

Store number is validated against the STORE table.

Numeric fields are checked for non-numeric characters.

For transaction of type SALE, RETURN and EEXCH, saimptlog checks whether a transaction is in balance:

- Transaction Items (Unit Retail \* Unit Retail Sign \* Quantity)
- + Item Discounts (Unit Discount Amount \* Unit Discount Sign \* Quantity)



- + Transaction Tax (Tax Amount \* Tax Sign)
  - = Transaction Tenders (Tender Amount \* Tender Sign)
  - saimptlog populates the Value field (on THEAD) with the transaction's sales value (item value – discount value + tax value) from the above calculation if it was not provided in the RTLOG.
6. Treatment of vouchers.
    - If an item sold is a gift certificate (Transaction Item, Voucher field has a value), issued information is written to the SA\_VOUCHER table.
    - If the Transaction Type is a RETURN, and the Transaction Tender Type Group is voucher (VOUCH), issued information is written to the SA\_VOUCHER table.
    - If the Transaction Type is a SALE, and the Transaction Tender Type Group is a voucher (VOUCH), redeemed information is written to the SA\_VOUCHER table.
    - When a gift certificate is sold, customer information should always be included. A receiving customer name value should be populated in the ref\_no5 field, a receiving customer state value should be populated in the ref\_no6 field and a receiving customer country should be populated in the ref\_no7 field. These reference fields can be changed by updating the sa\_reference table but the code needs to be modified too. The expiration date is put on the expiration\_date field on the TITEM record.
  7. Other validations/points of interest:
    - A salesperson in the TITEM record takes precedence over the salesperson in the THEAD record.
    - If an item sold is a sub-transaction (REF) item (Transaction Item, reference item field has a value and item does not), it is converted to the corresponding transaction level item (ITEM).
    - If an item sold is an ITEM (Transaction Item, item field has a value), it is validated against RMS item tables.
    - The corresponding Department, Class, Subclass, and Taxable Indicator is selected from the RMS tables and populated for an item.
  8. The balancing level determines whether the register or the cashier fields are required.
    - If the balancing level is 'R'egister, then the register field on the THEAD must be populated.
    - If the balancing level is 'C'ashier, then the cashier field on the THEAD must be populated.
    - If the balancing level is 'S'tore, then neither field is required to be populated.
  9. The tax\_ind and the item\_swiped\_ind fields can only accept 'Y' or 'N' values. If an invalid value is passed through the RTLOG, an error is flagged and the value is defaulted to 'Y'.

### Transaction of Type 'SALE'

A transaction of type SALE is generated whenever an item is sold. A sale may be to an employee, the sub-transaction type would be EMP in this case. Or it may be a drive-off sale (sub-transaction type DRIVEO) when someone drives off with unpaid gas. A special type of sale is an "odd exchange" (sub-transaction type EXCH) where items are sold and returned in the same transaction. If the net value of the exchange is positive, then it is a sale. If the net value is negative, it is a return.

Requirements per record type (other than what is described in Layout section above):

| Record Type | Requirements                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| THEAD       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| TITEM       | <ul style="list-style-type: none"> <li>Item Status is a required field; it determines whether the item is 'S'old, 'R'eturned or 'V'oided. If the item status is S, the quantity sign is expected to be P. If the item status is 'R', the quantity sign is expected to be N.</li> <li>If the item status is V, the quantity sign is the reverse of the quantity sign of the voided item. That is, if an item with status S is voided, the quantity sign would be N. Furthermore, the sum of the quantities being voided cannot exceed the sum of the quantities 'S'old or 'R'eturned. Note: neither of the above two validations are performed by saimptlog but an audit rule could be created to check this.</li> <li>In a typical sale, the items would all have a status of 'S'. In the case of an odd exchange, some items have a status of 'R'.</li> <li>In a typical return, the items would all have a status of 'R'. In the case of an odd exchange, some items have a status of 'S'.</li> <li>If an item has status R, then the Return Reason Code field may be populated. If it is, it is validated against code type 'SARR'.</li> <li>If the price of an item is overridden, then the Override Reason and Original Unit Retail fields must be populated.</li> </ul> |
| IDISC       | <ul style="list-style-type: none"> <li>The RMS Promotion Type field must always be populated with values of code type 'PRMT'.</li> <li>The Promotion field is validated, when a value is passed, against the promhead table.</li> <li>If the promotion is 'In Store' (code 1004), then the Discount Type field must be populated with values of code type 'SADT'.</li> <li>The Discount Reference Number is a promotion number which is of status 'A', 'E' or 'M'.</li> <li>If the Discount Type is 'SCOUP' for Store Coupon, then the Coupon Number field must be populated. The Coupon Reference Number field is optional.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| TTEND       | <ul style="list-style-type: none"> <li>If the tender type group is 'COUPON', then the Coupon Number field must be populated. The Coupon Reference Number field is optional.</li> <li>If the Transaction Tender Type Group is a credit card (CCARD), the number is validated against the SA_CC_VAL table. The other cc fields are optional.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |

Meaning of reference number fields:

**Note:** The meaning of these reference number fields may be changed through the sa\_reference table.

| Transaction Type | Sub-transaction Type | Item Type | Tender Type Group | Reference Number Field | Meaning of Reference Field | Req? |
|------------------|----------------------|-----------|-------------------|------------------------|----------------------------|------|
| SALE             |                      |           |                   | 1                      | Speed Sale Number          | Y    |
| SALE             |                      | GCN       |                   | 5                      | Recipient Name             | N    |
| SALE             |                      | GCN       |                   | 6                      | Recipient State            | N    |
| SALE             |                      | GCN       |                   | 7                      | Recipient Country          | N    |

| Transaction Type | Sub-transaction Type | Item Type | Tender Type Group | Reference Number Field | Meaning of Reference Field                           | Req? |
|------------------|----------------------|-----------|-------------------|------------------------|------------------------------------------------------|------|
| SALE             |                      |           | CHECK             | 9                      | Check Number                                         | N    |
| SALE             |                      |           | CHECK             | 10                     | Driver's License Number                              | N    |
| SALE             |                      |           | CHECK             | 11                     | Credit Card Number                                   | N    |
| SALE             | DRIVEO               |           |                   | 1                      | Incident Number                                      | Y    |
| SALE             | EMP                  |           |                   | 3                      | Employee Number of the employee receiving the goods. | N    |

Expected values for sign fields

| TRANSACTION TYPE | TITEM.Quantity Sign                                                                   | TTEND.Tender Sign | TTAX.Tax Sign | IDISC.Quantity Sign                                                                   |
|------------------|---------------------------------------------------------------------------------------|-------------------|---------------|---------------------------------------------------------------------------------------|
| SALE             | P if item is sold; N if item is returned; reverse of original item if item is voided. | P                 | P             | P if item is sold; N if item is returned; reverse of original item if item is voided. |

#### Transaction of Type 'PVOID'

This transaction is generated at the register when another transaction is being post voided. The orig\_tran\_no and orig\_reg\_no fields must be populated with the appropriate information for the transaction being post voided. The PVOID transaction must be associated with the same store day as the original transaction. If the PVOID needs to be generated after the store day is closed, the transaction needs to be created using the forms.

#### Transaction of Type 'RETURN'

This transaction is generated when a customer returns an item.

This type of transaction has similar record type requirements as a 'SALE' transaction.

Meaning of reference number fields:

**Note:** The meaning of these reference number fields may be changed through the sa\_reference table.

| Transaction Type | Sub-transaction Type | Reference Number Field | Meaning of Reference Field | Req? |
|------------------|----------------------|------------------------|----------------------------|------|
| RETURN           |                      | 1                      | Receipt Indicator (Y/N)    | Y    |
| RETURN           |                      | 2                      | Refund Reference Number    | N    |

| Transaction Type | Sub-transaction Type | Reference Number Field | Meaning of Reference Field                           | Req? |
|------------------|----------------------|------------------------|------------------------------------------------------|------|
| RETURN           | EMP                  | 3                      | Employee Number of the employee returning the goods. | N    |

Expected values for sign fields:

| TRANSACTION TYPE | TITEM.Quantity Sign                                                                   | TTEND.Tender Sign | TTAX.Tax Sign | IDISC.Quantity Sign                                                                   |
|------------------|---------------------------------------------------------------------------------------|-------------------|---------------|---------------------------------------------------------------------------------------|
| RETURN           | P if item is sold; N if item is returned; reverse of original item if item is voided. | N                 | N             | P if item is sold; N if item is returned; reverse of original item if item is voided. |

### Transaction of type 'EEXCH'

This transaction is generated when there is an even exchange.

This type of transaction has similar record type requirements as a 'SALE' transaction.

It is expected that the number of items returned equals the number of items sold.

However, this validation is not performed by saimptlog. An audit rule could be created for this. Saimptlog only expects that there would be at least two item records.

No tender changes hands in this transaction.

Meaning of reference number fields:

**Note:** The meaning of these reference number fields may be changed through the sa\_reference table.

| Transaction Type | Sub-transaction Type | Reference Number Field | Meaning of Reference Field                            | Req? |
|------------------|----------------------|------------------------|-------------------------------------------------------|------|
| EEXCH            |                      | 1                      | Receipt Indicator (Y/N)                               | Y    |
| EEXCH            | EMP                  | 3                      | Employee Number of the employee exchanging the goods. | N    |

### Transaction of Type 'PAIDIN'

This type of transaction has only one TTEND record.

A reason code is required.

Meaning of reference number fields:

**Note:** The meaning of these reference number fields may be changed through the sa\_reference table.

| Reason Code | Reference Number Column | Meaning                 | Req? |
|-------------|-------------------------|-------------------------|------|
| NSF         | 1                       | NFS Check Credit Number | N    |
| ACCT        | 1                       | Account Number          | N    |

### Transaction of Type 'PAIDOU'

This type of transaction has only one TTEND record.

A reason code is required (code type REAC). If the sub-transaction type is 'EV' or 'MV', the reason code comes from the non\_merch\_codes\_head table.

If the sub-transaction type is 'EV' or 'MV', then at least one field among the vendor number, vendor invoice number, payment reference number and proof of delivery number fields should be populated.

If the sub-transaction type is 'EV', then the vendor number comes from the partner table.

If the sub-transaction type is 'MV', then the vendor number comes from the supplier table.

Meaning of reference number fields:

**Note:** The meaning of these reference number fields may be changed through the sa\_reference table.

| Sub Transaction Type | Reason Code | Reference Number Column | Meaning            | Req? |
|----------------------|-------------|-------------------------|--------------------|------|
| EV                   |             | 2                       | Personal ID Number | N    |
| EV                   |             | 3                       | Routing Number     | N    |
| EV                   |             | 4                       | Account Number     | N    |
|                      | PAYRL       | 1                       | Money Order Number | N    |
|                      | PAYRL       | 2                       | Employee Number    | N    |
|                      | INC         | 1                       | Incident Number    | N    |

### Transaction of Type 'PULL'

This transaction is generated when cash is withdrawn from the register.

This type of transaction has only one TTEND record.

Expected values for sign fields:

| TRANSACTION TYPE | TITEM.Quantity Sign | TTEND.Tender Sign | TTAX.Tax Sign | IDISC.Quantity Sign |
|------------------|---------------------|-------------------|---------------|---------------------|
| PULL             | N/A                 | N                 | N/A           | N/A                 |

### Transaction of Type 'LOAN'

This transaction is generated when cash is added to the register.

This type of transaction has only one TTEND record.

Expected values for sign fields:

| TRANSACTION TYPE | TITEM.Quantity Sign | TTEND.Tender Sign | TTAX.Tax Sign | IDISC.Quantity Sign |
|------------------|---------------------|-------------------|---------------|---------------------|
| LOAN             | N/A                 | P                 | N/A           | N/A                 |

### Transaction of Type 'COND'

This transaction records the condition at the store when it opens. There can be at most one COND record containing weather information and at most one COND record containing temperature information. Both these pieces of information may be in the same COND record. There may be any number of COND records containing traffic and construction information.

This type of transaction does not have TITEM or IDISC or TTAX or TTEND records.

Meaning of reference number fields:

**Note:** The meaning of these reference number fields may be changed through the sa\_reference table.

| Reference Number Column | Meaning                                | Req? |
|-------------------------|----------------------------------------|------|
| 1                       | Weather – code type 'WEAT'             | N    |
| 2                       | Temperature – a signed 3 digit number. | N    |
| 3                       | Traffic – code_type 'TRAF'             | N    |
| 4                       | Construction – code_type 'CONS'        | N    |

### Transaction of Type 'TOTAL'

This transaction records the totals that are reported by the POS. The value field must be populated. Some POS systems generate only one transaction number for all totals. In order to avoid duplicate errors to be reported, only one total transaction can have a transaction number and the subsequent ones can have blank transaction numbers. In other words, a TOTAL transaction is not required to have a transaction number.

This type of transaction does not have TITEM or IDISC or TTAX or TTEND records.

### Transaction of Type 'METER'

This transaction is generated when a meter reading of a fuel pump is taken.

This type of transaction has only TITEM records.

Meaning of reference number fields:

**Note:** The meaning of these reference number fields may be changed through the sa\_reference table.

| Reference Number Column | Meaning                                                                             | Req? |
|-------------------------|-------------------------------------------------------------------------------------|------|
| 1                       | Reading Type: ('A' Adjustment, 'S' shift change, 'P' price change, 'C' store close) | Y    |

| Reference Number Column | Meaning                                                                                                              | Req? |
|-------------------------|----------------------------------------------------------------------------------------------------------------------|------|
| 5                       | Opening Meter Readings                                                                                               | Y    |
| 6                       | Closing Meter Reading                                                                                                | Y    |
| 7                       | If the reading type is 'P' for price change, the old unit retail should be placed here. Decimal places are required. | Y    |
| 8                       | Closing Meter Value                                                                                                  | Y    |

### Transaction of Type 'PUMPT'

This transaction is generated when a pump test is performed. This type of transaction has only TITEM records.

### Transactions of Type 'TANKDP'

This transaction is generated when a tank dip measurement is taken.

This type of transaction has only TITEM records.

Meaning of reference number fields:

**Note:** The meaning of these reference number fields may be changed through the sa\_reference table.

| Reference Number Column | Meaning                                    | Req? |
|-------------------------|--------------------------------------------|------|
| 1                       | Tank identifier                            | Y    |
| 5                       | Dip Type ('FUEL', 'WATER', etc.)           | Y    |
| 6                       | Dip Height Major (decimal places required) | Y    |
| 7                       | Dip Height Minor (decimal places required) | Y    |

### Transaction of Type 'DCLOSE':

This transaction is generated when day closed. Transaction number for this type of transaction has to be blank.

**Note:** Vouchers are minimally handled by saimptlog. Voucher information is written to the savouch file which is passed to the program savouch.pc.

- A voucher appears on the TITEM record only if it was sold. Thus when saimptlog encounters a 'SALE' transaction with a voucher, it writes the voucher to the savouch file as an 'I'ssued voucher.
- A voucher is issued when it appears on the TTEND record of transactions of type 'RETURN' and 'PAIDOU'. In other words, saimptlog writes it to the savouch file with status 'I'.
- A voucher is redeemed when it appears on the TTEND record of transactions of type 'SALE' and 'PAIDIN'. In other words, saimptlog writes it to the savouch file with status 'R'.

Vouchers may not be returned. However, a transaction of type 'PAIDOU' may be generated when the customer exchanges a voucher for another form of tender.

## saimptlogfin (Sales Audit Import)

### Functional Area

ReSA (Oracle Retail Sales Audit)

### Module Affected

SAIMPTLOGFIN.PC

### Design Overview

The SAIMPTLOGFIN program creates the balances (over or under) by store, register, or cashier and populates it in the SA\_BALANCE\_GROUP table, and it cancels post voided transactions and vouchers and validates missing transactions. It marks the store day record in the ReSA import log as partially or fully loaded. This unlocks the store day records after all store transactions are imported.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                    |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Sales Audit (Daily)                                                                                                                            |
| Scheduling Considerations | Run towards the end of loading POS transaction data into ReSA. It should run after SAIMPTLOG or SAIMPTLOGI and SAVOUCH but before SATOTALS.PC. |
| Pre-Processing            | N/A                                                                                                                                            |
| Post-Processing           | N/A                                                                                                                                            |
| Threading Scheme          | N/A                                                                                                                                            |

### Restart/Recovery

N/A

### Locking Strategy

This program is attempted to obtain a write lock on the store/day with a call to get\_lock library function. If this fails, go on to the next store/day and log the problem to the error log. If the lock is not obtained for a store/day then, skip the store\_day and process with the next store/day. It unlocks the store/day records if all transactions for the day are imported.

### Security Considerations

N/A

### Performance Considerations

N/A



**Key Tables Affected**

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| STORE_DAY_SEQ_NO       | Yes    | No     | No     | No     |
| STORE                  | Yes    | No     | No     | No     |
| SA_STORE_DAY           | Yes    | No     | Yes    | No     |
| SA_CUST_ATTRIB         | Yes    | No     | No     | Yes    |
| SA_CUSTOMER            | Yes    | No     | No     | Yes    |
| SA_TRAN_DISC           | Yes    | No     | No     | Yes    |
| SA_TRAN_ITEM           | Yes    | No     | No     | Yes    |
| SA_TRAN_TAX            | Yes    | No     | No     | Yes    |
| SA_TRAN_TENDER         | Yes    | No     | No     | Yes    |
| SA_ERROR               | Yes    | No     | Yes    | Yes    |
| SA_MISSING_TRAN        | Yes    | No     | No     | Yes    |
| SA_TRAN_HEAD           | Yes    | No     | Yes    | Yes    |
| SA_BALANCE_GROUP       | Yes    | Yes    | No     | No     |
| SA_TRAN_HEAD<br>CANCEL | Yes    | No     | No     | No     |
| SA_STORE_DATA          | Yes    | No     | No     | No     |
| SA_IMPORT_LOG          | No     | No     | Yes    | No     |
| SA_TRAN_HEAD_REV       | No     | Yes    | No     | No     |
| SA_TRAN_ITEM_REV       | No     | Yes    | No     | No     |
| SA_TRAN_TENDER_REV     | No     | Yes    | No     | No     |
| SA_TRAN_TAX_REV        | No     | Yes    | No     | No     |
| SA_TRAN_DISC_REV       | No     | Yes    | No     | No     |
| SA_ERROR_REV           | No     | Yes    | No     | No     |
| SA_EXPORTED_REV        | No     | Yes    | No     | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**sapreexp (Sales Audit Pre-Export)****Functional Area**

Oracle Retail Sales Audit (ReSA)

## Module Affected

SAPREEXP.PC

## Design Overview

When a user modifies or revises a transaction through the Oracle Retail Sales Audit user application, numerous totals are affected through re-totaling. The sales audit pre-export module is designed to compare the latest prioritized version of each total defined for export with the version that was previously sent to each system. If they are the same, an SA\_EXPORTED entry should be created for the total for that particular system so that the same value is not exported twice. By determining which totals have not changed since the last export date time (SA\_EXPORTED\_REV), this module then creates entries on SA\_EXPORTED to prohibit any third party application from receiving multiple export revisions.

## Scheduling Constraints

| Schedule Information      | Description                                                                                |
|---------------------------|--------------------------------------------------------------------------------------------|
| Processing Cycle          | Anytime                                                                                    |
| Scheduling Considerations | This module should be run after the ReSA auditing process and before any export processes. |
| Pre-Processing            | N/A                                                                                        |
| Post-Processing           | N/A                                                                                        |
| Threading Scheme          | N/A                                                                                        |

## Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Only two commits are done. One to establish the store/day lock (this is done by the package) and one at the end after a store/day or store/day/total is completely processed.

## Locking Strategy

This program is attempted to obtain a read lock on the store/day with a call to get\_lock function. If this fails, go on to the next store/day and log the problem to the error log.

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table         | Select | Insert | Update | Delete |
|---------------|--------|--------|--------|--------|
| SA_EXPORT_LOG | Yes    | No     | No     | No     |
| SA_STORE_DAY  | Yes    | No     | No     | No     |

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| SA_TOTAL_USAGE  | Yes    | No     | No     | No     |
| SA_EXPORT_LOG   | Yes    | No     | No     | No     |
| SA_EXPORTED     | Yes    | Yes    | No     | No     |
| SA_EXPORTED_REV | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**sapurge (Sales audit Purge)****Functional Area**

Oracle Retail Sale Audit (ReSA)

**Module Affected**

SAPURGE.PC

**Design Overview**

This program is run daily to control the size of the tables in the sales audit database. Older information is deleted to ensure optimal performance of the system as a whole. Different kinds of data need to be kept in the system for different amounts of time. Transactions, and all associated transaction details, and Totals calculated or reported for a store day is deleted when they meet the following criteria:

- The Business Date for those transactions and totals is older than or equal to today's date minus the days\_before\_purge parameter setup on the sa\_system\_options table.
- No locks exist on the store/day.
- One of the two following statements is true for the store/day:
  1. Fully loaded, and all errors either corrected or overridden (sa\_store\_day.audit\_status is 'A' (Audited) and sa\_store\_day.data\_status = 'F' (Fully loaded)). In addition, there are no outstanding exports (records for the store/day in the sa\_export\_log table where sa\_export\_log.status = 'R' (Ready for export)).
  2. Never loaded (sa\_store\_day.audit\_status is 'U' (Unaudited) and sa\_store\_day.data\_status = 'R' (Ready for import)).

Flash Sales data is deleted when it meets the following criteria:

- Date is two years before today's date minus the days\_before\_purge parameter setup on the sa\_system\_options.
- Company open and close dates also need to be kept for two years plus days\_before\_purge so that the historical comparisons in flash sales reporting carry the appropriate weight.

Voucher data is deleted when it meets the following criteria:

- The redeemed date or the escheat date for the specific voucher type is before today's date minus the purge\_no\_days on the sa\_voucher\_options table for the corresponding voucher type.

The program can also take in a list of store\_day\_seq\_no to delete. For example, the command line could be: sapurge userid/passwd 1000 1001 1002, where 1000, 1001 and 1003 are store\_day\_seq\_nos that the user wants to delete. These must also meet the criteria defined above. If a store\_day\_seq\_no is passed to this program, but does not meet the criteria, an error is written out to the error log.

An output file is created to store a record for each store and business date that was purged. The file name must be passed in at the command line as a parameter to sapurge.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                         |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Anytime – Sales Audit is a 24/7 system                                                                                              |
| Scheduling Considerations | This program should be run as the last program in the batch schedule. It can be run as part of the daily or monthly ReSA schedules. |
| Pre-Processing            | saprepost sapurge pre                                                                                                               |
| Post-Processing           | saprepost sapurge post                                                                                                              |
| Threading Scheme          | Threaded by store                                                                                                                   |

### Restart/Recovery

Restart/recovery is implicit in purge programs. The program only needs to be run again to restart appropriately.

### Locking Strategy

This program calls the get\_lock function which uses Oracle's DMBS\_LOCK functions to ensure exclusivity and avoid lock conflicts.

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| SA_STORE_DAY      | Yes    | No     | Yes    | Yes    |
| SA_SYSTEM_OPTIONS | Yes    | No     | No     | No     |
| SA_EXPORT_LOG     | Yes    | No     | No     | Yes    |
| SA_TRAN_HEAD_REV  | Yes    | No     | No     | Yes    |
| SA_TRAN_ITEM_REV  | Yes    | No     | No     | Yes    |

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| SA_TRAN_HEAD            | Yes    | No     | No     | Yes    |
| SA_TRAN_HEAD_TEMP       | Yes    | Yes    | No     | No     |
| SA_TOTAL                | Yes    | No     | No     | Yes    |
| SA_BALANCE_GROUP        | Yes    | No     | No     | Yes    |
| SA_ESCHEAT_TOTAL        | Yes    | No     | No     | Yes    |
| SA_VOUCHER_OPTIONS      | Yes    | No     | No     | No     |
| SA_EXPORTED             | No     | No     | No     | Yes    |
| SA_EXPORTED_REV         | No     | No     | No     | Yes    |
| SA_ERROR_REV            | No     | No     | No     | Yes    |
| SA_TRAN_TAX_REV         | No     | No     | No     | Yes    |
| SA_TRAN_DISC_REV        | No     | No     | No     | Yes    |
| SA_TRAN_TENDER_REV      | No     | No     | No     | Yes    |
| SA_TRAN_TAX             | No     | No     | No     | Yes    |
| SA_TRAN_DISC            | No     | No     | No     | Yes    |
| SA_TRAN_ITEM            | No     | No     | No     | Yes    |
| SA_TRAN_TENDER          | No     | No     | No     | Yes    |
| SA_CUST_ATTRIB          | No     | No     | No     | Yes    |
| SA_CUSTOMER             | No     | No     | No     | Yes    |
| SA_COMMENTS             | No     | No     | No     | Yes    |
| SA_ERROR                | No     | No     | No     | Yes    |
| SA_POS_VALUE            | No     | No     | No     | Yes    |
| SA_POS_VALUE_WKSHT      | No     | No     | No     | Yes    |
| SA_SYS_VALUE            | No     | No     | No     | Yes    |
| SA_SYS_VALUE_WKSHT      | No     | No     | No     | Yes    |
| SA_STORE_VALUE          | No     | No     | No     | Yes    |
| SA_HQ_VALUE             | No     | No     | No     | Yes    |
| SA_ERROR_WKSHT          | No     | No     | No     | Yes    |
| SA_MISSING_TRAN         | No     | No     | No     | Yes    |
| SA_IMPORT_LOG           | No     | No     | No     | Yes    |
| SA_BANK_ACH             | No     | No     | No     | Yes    |
| SA_ESCHEAT_VOUCHER      | No     | No     | No     | Yes    |
| SA_STORE_DAY_WRITE_LOCK | No     | No     | No     | Yes    |

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| SA_FLASH_SALES | No     | No     | No     | Yes    |
| SA_VOUCHER     | No     | No     | No     | Yes    |
| SA_STORE_ACH   | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**sarules (Sales Audit Rules)****Functional Area**

Oracle Retail Sales Audit (ReSA)

**Module Affected**

SARULES.PC

**Design Overview**

Evaluating rules is integral to the sales auditing process. Rules make the comparisons between data from various sources. These comparisons find data errors that could be the result of either honest mistakes or fraud. Finding these mistakes during the sales auditing process prevents these errors from being passed on to merchandising and data warehouse systems.

Rules in ReSA are dynamic. Rules are not predefined in the system. Retailers have the ability to define through the online Rule Definition Wizard. Errors uncovered by these rules are available for review during the interactive audit. As in the program, SATOTALS.PC, after users modify existing rules or create new ones, they become part of the rules the next time that SARULES.PC runs.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                                                                                                                                            |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Sales Audit – Runs daily                                                                                                                                                                                                                                               |
| Scheduling Considerations | This program runs after SATOTALS and before SAESCHEAT. It should also run before SAPREEXP and any of the ReSA export modules that extract store/day transaction data to other applications (for example, SAEXPRMS, SAEXPIM, SAEXPRDW, SAEXPACH, SAEXPUAR and SAEXPGL). |
| Pre-Processing            | N/A                                                                                                                                                                                                                                                                    |
| Post-Processing           | N/A                                                                                                                                                                                                                                                                    |
| Threading Scheme          | This program runs against a single store at a time.                                                                                                                                                                                                                    |

**Restart/Recovery**

The logical unit of work for this program is a SA\_STORE\_DAY record. Records are committed to the database when the commit\_max\_ctr defined for SARULES on the RESTART\_CONTROL table is reached. This program achieves inherent restart/recovery due to the fact that store/day records that are processed is updated to an audit\_status of 'A' (audited) or 'H' (HQ errors pending) or 'S' (store errors pending) and is not fetched by the driving cursor when the program restarts.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| SA_STORE_DAY      | Yes    | No     | Yes    | No     |
| SA_RULE_HEAD      | Yes    | No     | No     | No     |
| SA_RULE_LOC_TRAIT | Yes    | No     | No     | No     |
| SA_ERROR_WKSHT    | No     | Yes    | No     | Yes    |
| SA_ERROR_TEMP     | No     | Yes    | No     | No     |
| SA_ERROR          | No     | Yes    | Yes    | Yes    |
| SA_TOTAL          | No     | No     | Yes    | No     |
| SA_TRAN_HEAD      | No     | No     | Yes    | No     |
| SA_TRAN_ITEM      | No     | No     | Yes    | No     |
| SA_TRAN_DISC      | No     | No     | Yes    | No     |
| SA_TRAN_TENDER    | No     | No     | Yes    | No     |
| SA_TRAN_TAX       | No     | No     | Yes    | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**sastdyer (Sales Audit Store Day Create)****Functional Area**

Oracle Retail Sales Audit (ReSA)

**Module Affected**

SASTDYCR.PC

**Design Overview**

The SASTDYCR batch program creates store/day, imports log, and exports log records. This program should run prior to uploading the sales data from POS for a given store/day.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                              |
|---------------------------|----------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Daily – In the date set phase.                                                                           |
| Scheduling Considerations | It should run before the DTESYS batch program and before the next store/day's transactions are received. |
| Pre-Processing            | N/A                                                                                                      |
| Post-Processing           | N/A                                                                                                      |
| Threading Scheme          | N/A                                                                                                      |

**Restart/Recovery**

The logical unit of work in this program is store. Records are committed to the database when the commit counter is reached. The commit counter is defined by the value of INCREMENT\_BY on ALL\_SEQUENCE table for the sequence 'SA\_STORE\_DAY\_SEQ\_NO\_SEQUENCE.'

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| ALL_SEQUENCES        | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS       | Yes    | No     | No     | No     |
| STORE                | Yes    | No     | No     | No     |
| SA_STORE_DAY         | Yes    | Yes    | No     | No     |
| COMPANY_CLOSED       | Yes    | No     | No     | No     |
| COMPANY_CLOSED_EXCEP | Yes    | No     | No     | No     |
| LOCATION_CLOSED      | Yes    | No     | No     | No     |



| Table               | Select | Insert | Update | Delete |
|---------------------|--------|--------|--------|--------|
| PERIOD              | Yes    | No     | No     | No     |
| SA_IMPORT_LOG       | No     | Yes    | No     | No     |
| SA_EXPORT_LOG       | No     | Yes    | No     | No     |
| SA_FLASH_SALES      | No     | Yes    | No     | No     |
| SA_STORE_WRITE_LOCK | No     | Yes    | No     | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**satotals (Sales Audit Totals)****Functional Area**

Oracle Retail Sales Audit (ReSA)

**Module Affected**

SATOTALS.PC

**Design Overview**

This module produces totals from user-defined total calculation rules. Totaling is integral to the sales auditing process. Totaling provides the values against which auditors can compare receipts. These comparisons find data errors that could be the result of either honest mistakes or fraud. Finding these mistakes during the sales auditing process prevents these errors from being passed on to merchandising and data warehouse systems. Totaling also provides quick access to other numeric figures about the day's sales transactions.

Totaling in ReSA is dynamic. ReSA automatically totals transactions based on calculation definitions that the retailer's users create using the online Totals Calculation Definition Wizard. In addition, the retailer is able to define totals that come from the POS but that ReSA does not calculate. Whenever users create new calculation definitions or edit existing ones, they become part of the automated totaling process the next time that SATOTALS runs.

**Scheduling Constraints**

| Schedule Information      | Description                                                                       |
|---------------------------|-----------------------------------------------------------------------------------|
| Processing Cycle          | Sales Audit – Runs daily                                                          |
| Scheduling Considerations | This program runs after the batch programs SAIMPTLOGFIN.PC and before SARULES.PC. |
| Pre-Processing            | N/A                                                                               |

| Schedule Information | Description                                         |
|----------------------|-----------------------------------------------------|
| Post-Processing      | N/A                                                 |
| Threading Scheme     | This program runs against a single store at a time. |

### Restart/Recovery

The logical unit of work for this program is a SA\_STORE\_DAY record. Records are committed to the database when the commit\_max\_ctr defined for SATOTALS on the RESTART\_CONTROL table is reached. This program achieves inherent restart/recovery due to the fact that store/day records that are processed is updated to an audit\_status of 'T'otaled and is not be fetched by the driving cursor when the program restarts.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| SA_STORE_DAY       | Yes    | No     | Yes    | No     |
| SA_TOTAL           | No     | Yes    | No     | No     |
| SA_TOTAL_HEAD      | Yes    | No     | No     | No     |
| SA_ERROR           | No     | Yes    | No     | Yes    |
| SA_ERROR_WKSHT     | No     | Yes    | No     | Yes    |
| SA_POS_VALUE       | No     | Yes    | No     | No     |
| SA_POS_VALUE_WKSHT | No     | Yes    | No     | No     |
| SA_SYS_VALUE       | No     | Yes    | No     | No     |
| SA_SYS_VALUE_WKSHT | No     | Yes    | No     | No     |
| SA_ERROR_REV       | No     | Yes    | No     | No     |
| SA_EXPORTED_REV    | No     | Yes    | No     | No     |
| SA_EXPORTED        | No     | No     | No     | Yes    |

### Shared Modules

N/A

### I/O Specification

N/A

## savouch (Sales Audit Voucher Upload)

### Functional Area

Oracle Retail Sales Audit

### Module Affected

SAVOUCH.PC

### Design Overview

Because gift certificates can enter the Sales Audit system as either items or tender, processing must be done to match up the sales and redemptions. This module is used to aggregate gift certificate and voucher records. It compares records in the input files to the database. If a record for the voucher does not exist on the database, the record is inserted. If the voucher already exists on the database, the record should be updated with the appropriate information. The voucher details are updated to SA\_VOUCHER table.

Some retailers assign gift certificates to a given store, which means that before a gift certificate is sold at a store, it is assigned to a given store. When a retailer assigns a gift certificate to a given store, a record is written to the database. When the gift certificate is then sold by the store and redeemed by the consumer, this existing record must be updated to include the sale and redemption information. Some retailers choose not to assign gift certificates and instead simply sell gift certificates. In that case, the record is inserted into the database when the gift certificate is sold and then updated when the gift certificate is redeemed.

### Scheduling Constraints

| Schedule Information      | Description                                                                                            |
|---------------------------|--------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Sales Audit – Runs Daily                                                                               |
| Scheduling Considerations | This program needs to be scheduled after SAIMPTLOG.PC and its sqlldr process, but before saimptlogfin. |
| Pre-Processing            | N/A                                                                                                    |
| Post-Processing           | N/A                                                                                                    |
| Threading Scheme          | N/A                                                                                                    |

### Restart/Recovery

Restart/recovery logic for file based processing is used. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached.

### Locking Strategy

N/A

### Security Considerations

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table      | Select | Insert | Update | Delete |
|------------|--------|--------|--------|--------|
| SA_VOUCHER | Yes    | Yes    | Yes    | No     |

**I/O Specification****Input File Layout**

| Record Name | Field Name         | Field Type | Default Value | Description                                                                                                                   |
|-------------|--------------------|------------|---------------|-------------------------------------------------------------------------------------------------------------------------------|
| FHEAD       | Record descriptor/ | Char(5)    | FHEAD         | File head marker                                                                                                              |
|             | Line id            | Number(10) | 0000000001    | Unique line id                                                                                                                |
|             | Translator id      | Char(5)    | SAVO          | Identifies transaction type                                                                                                   |
|             | File create date   | Char(14)   |               | Vdate in YYYYMMDDHH24MISS format                                                                                              |
|             | Business Date      | Char(8)    | Business Date | Vdate in YYYYMMDD format                                                                                                      |
| FDETL       | Record descriptor/ | Char(5)    | FDETL         | File head marker                                                                                                              |
|             | Line id            | Number(10) |               | Unique line id                                                                                                                |
|             | Voucher seq Number | Number(20) |               | Unique identifier for an entry to SA_VOUCHER table                                                                            |
|             | Voucher No         | Char(16)   |               | Serial Number of the voucher                                                                                                  |
|             | Tender Type Id     | Number(6)  |               | Type of Voucher (Valid values for tender type are maintained in pos_tender_type_head table with tender_type_group as 'VOUCH') |
|             | Assigned Date      | Char(8)    |               | Date the voucher was assigned                                                                                                 |
|             | Assigned store     | Number(10) |               | Store to which the voucher is assigned                                                                                        |
|             | Issuing Date       | Char(8)    |               | Date this document was issued                                                                                                 |
|             | Issuing store      | Number(10) |               | Store this document was issued from                                                                                           |
|             | Issuing Register   | Char(5)    |               | Register this document was issued from                                                                                        |
|             | Issuing Cashier    | Char(10)   |               | Cashier issuing the document                                                                                                  |

| Record Name | Field Name                  | Field Type | Default Value | Description                                                                                        |
|-------------|-----------------------------|------------|---------------|----------------------------------------------------------------------------------------------------|
|             | Issued transaction number   | Number(20) |               | Transaction number at the time of issuance                                                         |
|             | Issued item seq number      | Number(4)  |               | Holds the item sequence of the item when the voucher is sold as an item (gift voucher)             |
|             | Issued tender seq number    | Number(4)  |               | Holds the tender sequence of the tender when the voucher is sold as a tender (Merchandise Credit). |
|             | Issued Amount               | Number(20) |               | Amount the document was issued for * 10000 (4 implied decimal places)                              |
|             | Issued Customer Name        | Char(120)  |               | Name of the customer, who was issued the voucher                                                   |
|             | Issued Customer Addr1       | Char(240)  |               | The address of the customer who was issued the voucher                                             |
|             | Issued Customer Addr2       | Char(240)  |               | The second line address of the customer who was issued the voucher                                 |
|             | Issued Customer City        | Char(120)  |               | City of the customer, the voucher is issued                                                        |
|             | Issued Customer State       | Char(3)    |               | State of the customer                                                                              |
|             | Issued Customer Postal Code | Char(30)   |               | Postal address of the customer.                                                                    |
|             | Issued Customer Country     | Char(3)    |               | Country of the customer the voucher was issued.                                                    |
|             | Recipient Name              | Char(120)  |               | Name of the intended recipient                                                                     |
|             | Recipient State             | Char(3)    |               | The state of the intended recipient.                                                               |
|             | Recipient Country           | Char(3)    |               | The country of the intended recipient.                                                             |
|             | Redemption Date             | Char(8)    |               | Date the voucher was redeemed.                                                                     |
|             | Redemption Store            | Number(10) |               | Store, the voucher was redeemed at.                                                                |

| Record Name | Field Name                   | Field Type | Default Value | Description                                                                                                     |
|-------------|------------------------------|------------|---------------|-----------------------------------------------------------------------------------------------------------------|
|             | Redemption Register          | Char(5)    |               | Register, the document was redeemed at.                                                                         |
|             | Redemption cashier           | Char(10)   |               | Cashier redeeming the voucher                                                                                   |
|             | Redemption tran seq number   | Number(20) |               | Transaction Number when the document was redeemed                                                               |
|             | Redemption Tender seq number | Number(4)  |               | This column holds the tender sequence of the tender within the transaction when a voucher is redeemed as tender |
|             | Redemption Amount            | Number(20) |               | Amount the document was redeemed for * 10000 (4 implied decimal places)                                         |
|             | Expiry Date                  | Char(8)    |               | Expiry Date                                                                                                     |
|             | Status                       | Char(1)    |               | Indicator showing the document's status, issued or redeemed. Valid values = I – Issued, R - Redeemed.           |
|             | Comments                     | Char(2000) |               | Comments                                                                                                        |
| FTAIL       | Record type                  | Char(5)    | FTAIL         | Describes file record – marks end of file                                                                       |
|             | Line id                      | Number(10) |               | Unique file line ID                                                                                             |
|             | #lines                       | Number(10) |               | Total number of transaction lines in file (not including FHEAD and FTAIL)                                       |

## Oracle Retail Trade Management Batch

### Overview

Oracle Retail Trade Management (RTM) automates international import transaction data. There are six components of RTM:

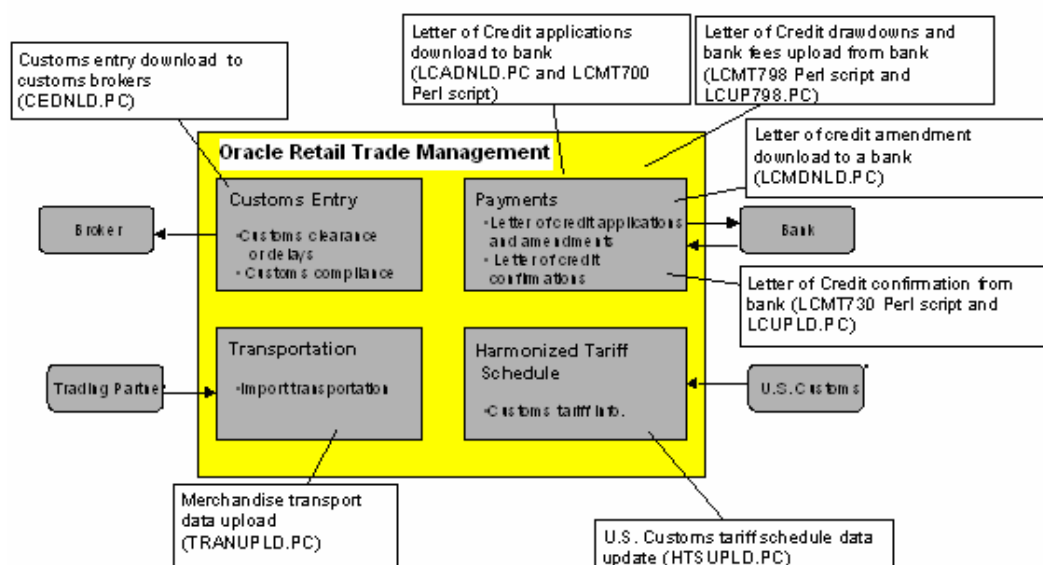
- Customs entry
- Harmonized tariff schedule
- Letter of credit
- Transportation.
- Actual landed costs
- Obligations

Four of these components—customs entry, Harmonized Tariff Schedule, letter of credit, and transportation—have batch-processing modules that facilitate the flow of data between RTM and external applications and files. This document describes these batch modules, along with Perl scripts, and the kinds of data that they process.

### Invoice and Accounts Payable Integration

Obligations and customs entry costs can be approved for payment and entered into Oracle Retail Invoice Matching (ReIM). Invoice data can then be sent to the retailer's financial application for payment.

The following graphic describes RTM functionality and batch processing.



### RTM Functionality and Batch Processing

## Batch Design Summary

The following batch designs are included in this functional area:

- CEDNLD.PC (Customs Entry Download)
- HTSUPLD.PC (Harmonized Tariff Schedule Upload)

### Letter of Credit

- LCADNLD.PC (Letter of Credit Application Download)
- LCMDNLD.PC (Letter of Credit Amendment Download)
- LCMT700 Perl script (SWIFT File Conversion)
- LCMT707 (Converts Letter of Credit from Oracle Retail Format to SWIFT)
- LCMT730 Perl script (SWIFT File Conversion – Letter of Credit Information)
- LCMT798 Perl script (SWIFT File Conversion – Letter of Credit Changes and Drawdowns)
- LCUP798.PC (Letter of Credit Upload for S.W.I.F.T. Format 798)
- LCUPLD.PC (Letter of Credit Upload)
- TRANUPLD.PC (Transportation Upload)

## cednld (Customs Entry Download)

### Functional Area

Customs entry download

### Module Affected

CEDNLD.PC

### Design Overview

This program is used to download custom entry information from the RMS database to brokers. Each night, this program reads all custom entry (CE) transactions that are in a “S”ent status for a broker ID. These transactions are written to a flat file and the status is changed to “D”ownloaded. One process runs, and one flat file is written per broker.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                  |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 2 (daily)                                                                                                                              |
| Scheduling Considerations | N/A                                                                                                                                          |
| Pre-Processing            | N/A                                                                                                                                          |
| Post-Processing           | N/A                                                                                                                                          |
| Threading Scheme          | Single Threaded, however multiple processes can be run at the same time, each downloading customer entry information for a different broker. |



## Restart/Recovery

The Logical Unit of Work for the program is a single row from the CE\_HEAD table. Restart/Recovery is used for init and commit.

Table based restart/recovery must be used. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 1000 records (subject to change based on implementation).

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| CE_HEAD         | Yes    | No     | Yes    | No     |
| CE_SHIPMENT     | Yes    | No     | No     | No     |
| CE_ORD_ITEM     | Yes    | No     | No     | No     |
| ORDHEAD         | Yes    | No     | No     | No     |
| SUP_IMPORT_ATTR | Yes    | No     | No     | No     |
| TRANSPORTATION  | Yes    | No     | No     | No     |
| CE_LIC_VISA     | Yes    | No     | No     | No     |
| CE_CHARGES      | Yes    | No     | No     | No     |
| MISSING_DOC     | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name           | Field Type | Default Value                      | Description                                       |
|-------------|----------------------|------------|------------------------------------|---------------------------------------------------|
| File Header | File Type Descriptor | Char(5)    | FHEAD                              | Identifies file record type                       |
|             | File Line Identifier | Number(10) | Nine leading zeroes:<br>0000000001 | ID of current line being processed by input file. |
|             | File Type Definition | Char(4)    | CEDN                               | Identifies file as 'Customs Entry download'       |

| Record Name | Field Name           | Field Type     | Default Value                                                     | Description                                       |
|-------------|----------------------|----------------|-------------------------------------------------------------------|---------------------------------------------------|
|             | File Create Date     | Date           | Create date                                                       | Vdate in YYYYMMDDHH24MISS format.                 |
| CE_HEAD     | File Type Descriptor | Char(5)        | THEAD                                                             | Identifies file record type                       |
|             | File Line Identifier | Number(10)     | Incremented internally                                            | ID of current line being processed by input file. |
|             | CE ID                | Number(10)     | ce_head.ce_id                                                     |                                                   |
|             | Entry No             | Char (15)      | ce_head.entry_no                                                  |                                                   |
|             | Entry Date           | Char(14)       | ce_head.entry_date                                                | YYYYMMDDHH24MISS format                           |
|             | Entry Status         | Char(6)        | ce_head.entry_status                                              |                                                   |
|             | Entry Type           | Char(6)        | ce_head.entry_type                                                |                                                   |
|             | Entry Port           | Char(5)        | ce_head.entry_port                                                |                                                   |
|             | Summary Date         | Char(14)       | ce_head.summary_date                                              | YYYYMMDDHH24MISS format                           |
|             | Broker ID            | Char(10)       | ce_head.broker_id                                                 |                                                   |
|             | Broker Ref. ID       | Char(18)       | ce_head.broker_ref_id                                             |                                                   |
|             | File Number          | Char(18)       | ce_head.file_no                                                   |                                                   |
|             | Importer ID          | Char(10)       | ce_head.importer_id                                               |                                                   |
|             | Import Country       | Char(3)        | ce_head.import_country_id                                         |                                                   |
|             | Currency Code        | Char(3)        | ce_head.currency_code                                             |                                                   |
|             | Exchange Rate        | Number(20, 10) | ce_head.exchange_rate*1000000000 (with 10 implied decimal places) |                                                   |
|             | Bond Number          | Char(18)       | ce_head.bond_no                                                   |                                                   |
|             | Bond Type            | Char(6)        | ce_head.bond_type                                                 |                                                   |

| Record Name | Field Name               | Field Type    | Default Value                                              | Description                                       |
|-------------|--------------------------|---------------|------------------------------------------------------------|---------------------------------------------------|
|             | Surety Code              | Char(6)       | ce_head.surety_code                                        |                                                   |
|             | Consignee ID             | Char(10)      | ce_head.consignee_id                                       |                                                   |
|             | Live Indicator           | Char(1)       | ce_head.live_ind                                           |                                                   |
|             | Batch Number             | Char(20)      | ce_head.batch_no                                           |                                                   |
|             | Entry Team               | Char(3)       | ce_head.entry_team                                         |                                                   |
|             | Liquidation Amount       | Number(20, 4) | ce_head.liquidation_amt*10000 (4 implied decimal places)   |                                                   |
|             | Liquidation Date         | Date          | ce_head.liquidation_date                                   | YYYYMMDDHH24MISS format                           |
|             | Reliquidation Amount     | Number(20, 4) | ce_head.reliquidation_amt*10000 (4 implied decimal places) |                                                   |
|             | Reliquidation Date       | Date          | ce_head.reliquidation_date                                 | YYYYMMDDHH24MISS format                           |
|             | Merchandise Loc          | Char(40)      | ce_head.merchandise_loc                                    |                                                   |
|             | Location Code            | Char(4)       | ce_head.location_code                                      |                                                   |
| CE_SHIPMENT | File Type Descriptor     | Char(5)       | TSHIP                                                      | Identifies file record type                       |
|             | File Line Identifier     | Number(10)    | Incremented internally                                     | ID of current line being processed by input file. |
|             | Vessel ID                | Char(20)      | ce_shipment.vessel_id                                      |                                                   |
|             | Voyage Flt ID            | Char(10)      | ce_shipment.voyage_flt_id                                  |                                                   |
|             | Estimated Departure Date | Date          | ce_shipment.estimated_depart_date                          | YYYYMMDDHH24MISS format                           |

| Record Name | Field Name           | Field Type    | Default Value                                            | Description                                                                         |
|-------------|----------------------|---------------|----------------------------------------------------------|-------------------------------------------------------------------------------------|
|             | Vessel SCAC Code     | Char(6)       | ce_shipment.vessel_scac_code                             |                                                                                     |
|             | Lading Port          | Char(5)       | ce_shipment.lading_port                                  |                                                                                     |
|             | Discharge Port       | Char(5)       | ce_shipment.discharge_port                               |                                                                                     |
|             | Tran Mode ID         | Char(6)       | ce_shipment.tran_mode_id                                 |                                                                                     |
|             | Export Date          | Date          | ce_shipment.export_date                                  | YYYYMMDDHH24MISS                                                                    |
|             | Import Date          | Date          | ce_shipment.import_date                                  | YYYYMMDDHH24MISS                                                                    |
|             | Arrival Date         | Date          | ce_shipment.arrival_date                                 | YYYYMMDDHH24MISS                                                                    |
|             | Export Country       | Char(3)       | ce_shipment.export_country_id                            |                                                                                     |
|             | Shipment Number      | Number(10)    | ce_shipment.shipment_no                                  |                                                                                     |
| CE_ORD_ITEM | File Type Descriptor | Char(5)       | TORDI                                                    | Identifies file record type                                                         |
|             | File Line Identifier | Number(10)    | Incremented internally                                   | ID of current line being processed by input file.                                   |
|             | Order Number         | Number(8)     | ce_ord_item.order_no                                     |                                                                                     |
|             | Item                 | Char (25)     | ce_ord_item.item                                         |                                                                                     |
|             | BL AWB ID            | Char(30)      | ce_ord_item.bl_awb_id                                    | 'MULTI' – means multiple airway bills (otherwise a single airway bill is retrieved) |
|             | Invoice ID           | Char(30)      | ce_ord_item.invoice_id                                   |                                                                                     |
|             | Invoice Date         | Date          | ce_ord_item.invoice_date                                 | YYYYMMDDHH24MISS format                                                             |
|             | Invoice Amount       | Number(20, 4) | ce_ord_item.invoice_amt*10000 (4 implied decimal places) |                                                                                     |

| Record Name | Field Name                 | Field Type     | Default Value                                                                | Description |
|-------------|----------------------------|----------------|------------------------------------------------------------------------------|-------------|
|             | Currency Code              | Char(3)        | ce_ord_item.curren<br>ncy_code                                               |             |
|             | Exchange Rate              | Number(20, 10) | ce_ord_item.excha<br>nge_rate*1000000<br>0000 (10 implied<br>decimal places) |             |
|             | Manifest Item Quantity     | Number(12, 4)  | ce_ord_item.manif<br>est_item_qty*1000<br>0 (4 implied<br>decimal places)    |             |
|             | Manifest Item Quantity UOM | Char(4)        | ce_ord_item.manif<br>est_item_qty_uom                                        |             |
|             | Carton Quantity            | Number (12,4)  | ce_ord_item.carto<br>n_qty*10000 (4<br>implied decimal<br>places)            |             |
|             | Carton Quantity UOM        | Char(4)        | ce_ord_item.carto<br>n_qty_uom                                               |             |
|             | Gross Weight               | Number(12, 4)  | ce_ord_item.gross<br>_wt*10000 (4<br>implied decimal<br>places)              |             |
|             | Gross Weight UOM           | Char(4)        | ce_ord_item.gross<br>_wt_uom                                                 |             |
|             | Net Weight                 | Number(12, 4)  | ce_ord_item.net_w<br>t*10000 (4 implied<br>decimal places)                   |             |
|             | Net Weight UOM             | Char(4)        | ce_ord_item.net_w<br>t_uom                                                   |             |
|             | Cubic                      | Number(12, 4)  | ce_ord_item.cubic<br>*10000 (4 implied<br>decimal places)                    |             |
|             | Cubic UOM                  | Char(4)        | ce_ord_item.cubic<br>_uom                                                    |             |

| Record Name | Field Name           | Field Type    | Default Value                                            | Description                                       |
|-------------|----------------------|---------------|----------------------------------------------------------|---------------------------------------------------|
|             | Cleared Quantity     | Number(12, 4) | ce_ord_item.cleared_qty*10000 (4 implied decimal places) |                                                   |
|             | Cleared Quantity UOM | Char(4)       | ce_ord_item.cleared_qty_uom                              |                                                   |
|             | In Transit Number    | Char(15)      | ce_ord_item.in_transit_no                                |                                                   |
|             | In Transit Date      | Date          | ce_ord_item.in_transit_date                              | YYYYMMDDHH24MISS format                           |
|             | Rush Indicator       | Char(1)       | ce_ord_item.rush_ind                                     |                                                   |
|             | Related Indicator    | Char(1)       | ce_ord_item.related_ind                                  |                                                   |
|             | Tariff Treatment     | Char(10)      | ce_ord_item.tariff_treatment                             |                                                   |
|             | Ruling Number        | Char(10)      | ce_ord_item.ruling_no                                    |                                                   |
|             | Do Number            | Char(10)      | ce_ord_item.do_no                                        |                                                   |
|             | Do Date              | Date          | ce_ord_item.do_date                                      | YYYYMMDDHH24MISS format                           |
|             | Manufacture ID       | Char(18)      | sup_import_attr.mfg_id                                   |                                                   |
| BL_AWB_ID   | File Type Descriptor | Char(5)       | TBLAW                                                    | Identifies file record type                       |
|             | File Line Identifier | Number(10)    | Incremented internally                                   | ID of current line being processed by input file. |
|             | BL AWB ID            | Char(30)      | Transportation.bl_awb_id                                 |                                                   |
| CONTAINER   | File Type Descriptor | Char(5)       | TCONT                                                    | Identifies file record type                       |
|             | File Line Identifier | Number(10)    | Incremented internally                                   | ID of current line being processed by input file. |
|             | Container ID         | Char(20)      | Transportation.container_id                              |                                                   |

| Record Name | Field Name                | Field Type    | Default Value                                                 | Description                                       |
|-------------|---------------------------|---------------|---------------------------------------------------------------|---------------------------------------------------|
|             | Container SCAC Code       | Char(6)       | Transportation.container_scac_code                            |                                                   |
| CE_LIC_VISA | File Type Descriptor      | Char(5)       | TLICV                                                         | Identifies file record type                       |
|             | File Line Identifier      | Number(10)    | Incremented internally                                        | ID of current line being processed by input file. |
|             | License/Visa Type         | Char(6)       | ce_lic_visa.license_visa_type                                 |                                                   |
|             | License/Visa ID           | Char(30)      | ce_lic_visa.license_visa_id                                   |                                                   |
|             | License/Visa Quantity     | Number(12, 4) | ce_lic_visa.license_visa_qty*10000 (4 implied decimal places) |                                                   |
|             | License/Visa Quantity UOM | Char(4)       | ce_lic_visa.license_visa_qty_uom                              |                                                   |
|             | Quota Category            | Char (6)      | ce_lic_visa.quota_category                                    |                                                   |
|             | Net Weight                | Number(12, 4) | ce_lic_visa.net_weight*10000 (4 implied decimal places)       |                                                   |
|             | Net Weight UOM            | Char(4)       | ce_lic_visa.net_weight_uom                                    |                                                   |
|             | Holder ID                 | Char(18)      | ce_lic_visa.holder_id                                         |                                                   |
| CE_CHARGES  | File Type Descriptor      | Char(5)       | TCHRG                                                         | Identifies file record type                       |
|             | File Line Identifier      | Number(10)    | Incremented internally                                        | ID of current line being processed by input file. |
|             | Sequence Number           | Number(6)     | ce_charges.seq_no                                             |                                                   |
|             | Pack Item                 | char(25)      | ce_charges.pack_item                                          |                                                   |

| Record Name  | Field Name           | Field Type    | Default Value                                            | Description                                                                                 |
|--------------|----------------------|---------------|----------------------------------------------------------|---------------------------------------------------------------------------------------------|
|              | HTS                  | Char(10)      | ce_charges.hts                                           |                                                                                             |
|              | Effect From Date     | Date          | ce_charges.effect_from                                   | YYYYMMDDHH24MISS format                                                                     |
|              | Effect To Date       | Char(14)      | ce_charges.effect_to                                     | YYYYMMDDHH24MISS format                                                                     |
|              | Component ID         | Date          | ce_charges.comp_id                                       |                                                                                             |
|              | Component Rate       | Number(20, 4) | ce_charges.comp_rate*10000 (4 implied decimal places)    |                                                                                             |
|              | Per Count UOM        | Char(3)       | ce_charges.per_count_uom                                 |                                                                                             |
|              | Component Value      | Number(20, 4) | ce_charges.comp_value * 10000 (4 implied decimal places) |                                                                                             |
| MISSING_DOC  | File Type Descriptor | Char(5)       | TMDOC                                                    | Identifies file record type                                                                 |
|              | File Line Identifier | Number(10)    | Incremented internally                                   | ID of current line being processed by input file.                                           |
|              | Doc_id               | Number(6)     | Missing_doc.doc_id                                       |                                                                                             |
|              | Received_date        | Date          | Missing_doc.received_date                                | YYYYMMDDHH24MISS format                                                                     |
| File Trailer | File Type Descriptor | Char(5)       | FTAIL                                                    | Identifies file record type                                                                 |
|              | File Line Identifier | Number(10)    | Incremented internally                                   | ID of current line being processed by input file.                                           |
|              | File Record Counter  | Number(10)    | Determined Internally                                    | Number of records/transactions processed in current file (only records between head & tail) |

## htsupld (Harmonized Tariff Schedule Upload)

### Functional Area

Trade Management batch



**Module Affected**

HTSUPLD.PC

**Design Overview**

The harmonized tariff schedule HTSUPLD.PC module processes a file containing the most recent United States Customs tariff schedule to RMS tables. The module uploads both the initial entry of the schedule and all the updates, as they become available.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                                                                                                                                                    |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Ad Hoc Interfaces                                                                                                                                                                                                                                                              |
| Scheduling Considerations | Run any time as needed                                                                                                                                                                                                                                                         |
| Pre-Processing            | Hts240_to_2400 (perl script to convert the original US government HTS file of 240-char lines to 2400-char lines)<br><br>Ushts2rms (perl script to convert the HTS file of 2400-char lines to standard Oracle Retail file format)<br><br>PREPOST.PC with HTSUPLD_PRE() function |
| Post-Processing           | N/A                                                                                                                                                                                                                                                                            |
| Threading Scheme          | The number of threads are based on the number of input files.                                                                                                                                                                                                                  |

**Restart/Recovery**

Recommended commit counter is 2000. Input file names must end in a “.1” for the restart mechanism to properly parse the file name. Because there is only 1 input file to be uploaded, only 1 thread is used.

A reject file is used to hold records that have failed processing. The user can fix the rejected records and process the reject file again.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| HTS                  | Yes    | Yes    | Yes    | Yes    |
| HTS_TARIFF_TREATMENT | Yes    | Yes    | Yes    | Yes    |

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| ITEM HTS               | Yes    | Yes    | Yes    | Yes    |
| MOD_ORDER_ITEM HTS     | No     | Yes    | No     | No     |
| HTS_OGA                | No     | Yes    | Yes    | Yes    |
| ORDSKU HTS             | Yes    | Yes    | Yes    | Yes    |
| HTS_TT_EXCLUSIONS      | No     | Yes    | Yes    | Yes    |
| HTS_TAX                | No     | Yes    | Yes    | Yes    |
| HTS_FEE                | No     | Yes    | Yes    | Yes    |
| CE_CHARGES             | Yes    | Yes    | Yes    | Yes    |
| HTS_CHAPTER            | Yes    | Yes    | No     | No     |
| QUOTA_CATEGORY         | Yes    | Yes    | No     | No     |
| ITEM HTS_ASSESS        | No     | Yes    | Yes    | Yes    |
| HTS_AD                 | No     | No     | Yes    | No     |
| HTS_CVD                | No     | No     | Yes    | No     |
| HTS_REFERENCE          | No     | No     | Yes    | No     |
| ORDHEAD                | Yes    | No     | Yes    | No     |
| ITEM_EXP_DETAIL        | No     | No     | Yes    | No     |
| ORDLOC_EXP             | No     | No     | Yes    | No     |
| ORDSKU HTS_ASSESS      | No     | No     | Yes    | Yes    |
| ORDSKU_TEMP            | Yes    | No     | No     | Yes    |
| ORDLOC_TEMP            | No     | No     | No     | Yes    |
| ALLOC_CHRG_TEMP        | No     | No     | No     | Yes    |
| ALLOC_DETAIL_TEMP      | No     | No     | No     | Yes    |
| ALLOC_HEADER_TEMP      | No     | No     | No     | Yes    |
| ORDLOC_EXP_TEMP        | No     | No     | No     | Yes    |
| ORDSKU HTS_ASSESS_TEMP | No     | No     | No     | Yes    |
| ORDSKU HTS_TEMP        | No     | No     | No     | Yes    |
| ORDLOC_DISCOUNT_TEMP   | No     | No     | No     | Yes    |
| TIMELINE_TEMP          | No     | No     | No     | Yes    |
| REQ_DOC_TEMP           | No     | No     | No     | Yes    |
| WO_DETAIL_TEMP         | No     | No     | No     | Yes    |
| WO_HEAD_TEMP           | No     | No     | No     | Yes    |
| REPL_RESULTS_TEMP      | No     | No     | No     | Yes    |

## I/O Specification

### Input File Layout

| Record Name | Field Name        | Field Type | Default Value | Description                                                                    |
|-------------|-------------------|------------|---------------|--------------------------------------------------------------------------------|
| FHEAD       | Record Descriptor | Char(5)    | FHEAD         | Describes file line type                                                       |
|             | Line number       | Number(10) | 0000000001    | Sequential file line number                                                    |
|             | Retek file ID     | Char(5)    | HTSUP         | Describes file type                                                            |
| THEAD       | Record Descriptor | Char(5)    | THEAD         | Describes file line type                                                       |
|             | Line number       | Number(10) |               | Sequential file line number                                                    |
|             | Transaction id    | Number(14) |               | Unique transaction id                                                          |
|             | HTS Line          | Char(358)  |               | V1 through V4 records from the customs HTS file concatenated together          |
| TDETL       | Record Descriptor | Char(5)    | TDETL         | Describes file line type                                                       |
|             | Line number       | Number(10) |               | Sequential file line number                                                    |
|             | Transaction id    | Number(10) |               | Unique transaction id                                                          |
|             | Tax/fee line      | Char(80)   |               | V5 through VC records from the customs HTS file, each on a separate TDETL line |
| TTAIL       | Record Descriptor | Char(5)    | TTAIL         | Describes file line type                                                       |
|             | Line number       | Number(10) |               | Sequential file line number                                                    |
|             | Detail lines      | Number(6)  |               | Number of lines between THEAD and TTAIL                                        |
| FTAIL       | Record Descriptor | Char(5)    | FTAIL         | Describes file line type                                                       |
|             | Line number       | Number(10) |               | Sequential file line number                                                    |
|             | Transaction Lines | Number(10) |               | Number of lines between FHEAD and FTAIL                                        |

**Here is the layout of the original input file:**

**Note:** The input file contains lines of 2400 characters (that is, the newline character occurs only after every 2400 characters). Each 2400-character line consists of thirty 80-character records. Each 80-character record starts with 'V1' or 'V2' ... or 'VD' or blank if the record is completely empty. For each tariff, records V1 and V2 are mandatory; records V3 through VD are optional, which means they can be all blank. Record V4 is not currently used in RMS/RTM. Records V5 through VC contain the tax/fee information for the tariff, and all have the same structure. The lower-case letters in the record name block are as a convenience to cross-reference with the US Customs file description.

| Record Name | Field Name           | Field Type | Default Value | Description                                                                                                                                                                                 |
|-------------|----------------------|------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| V1 a        | Control identifier   | Char(1)    | V             | Identifies start of record                                                                                                                                                                  |
| b           | Record type          | Char(1)    | 1             | Identifies record type                                                                                                                                                                      |
| c           | Tariff number        | Number(10) |               | A code located in the <i>Harmonized Tariff Schedule of the United States Annotated</i> (HTS) representing the tariff number. If this number is less than 10 positions, it is left justified |
| d           | Transaction code     | Char(1)    | A, D, R       | A code representing the type of transaction. Valid Transaction Codes are:<br><br>A = Add<br>D = Delete<br>R = Replace                                                                       |
| e           | Begin effective date | char(6)    |               | A numeric date in MMDDYY (month, day, year) format representing the record begin effective date. This date indicates when the record becomes effective.                                     |
| f           | End effective date   | char(6)    |               | A numeric date in MMDDYY (month, day, year) format representing the record end effective date. This date indicates the last date the record is effective.                                   |

| Record Name | Field Name                                | Field Type | Default Value | Description                                                                                                                                                                                                                                                                       |
|-------------|-------------------------------------------|------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| g           | number of reporting units                 | number(1)  | 0,1,or 2 or 3 | The number of reporting units required by the Bureau of the Census. In a few instances, units not required by Census may be required to compute duty. In these cases, the Census reporting units are always first, followed by any additional units required to compute the duty. |
| h           | 1 <sup>st</sup> reporting unit of measure | char(4)    |               | A code representing the first unit of measure. If the reporting unit is X, no unit of measure is required except for certain tariff numbers in Chapter 99. Valid unit of measure codes are listed in Appendix C.                                                                  |
| I           | 2 <sup>nd</sup> reporting unit of measure | char(4)    |               | A code representing the second unit of measure. Valid unit of measure codes are listed in Appendix C.                                                                                                                                                                             |
| j           | 3 <sup>rd</sup> reporting unit of measure | char(4)    |               | A code representing the third unit of measure. Valid unit of measure codes are listed in Appendix C.                                                                                                                                                                              |
| k           | duty computation code                     | char(1)    |               | A code indicating the formula to be used to compute the duty. Valid Duty Computation Codes are listed in Appendix F.                                                                                                                                                              |
| l           | commodity description                     | char(30)   |               | A condensed version of the commodity description that appears in the HTS.                                                                                                                                                                                                         |
| m           | column 1 specific rate of duty            | Number(12) |               | The rate of duty that appears in the General column of the HTS. Eight decimal places are implied.                                                                                                                                                                                 |
| n           | base rate indicator                       | char(1)    | 'B' or blank  | A code indicating if the rate contains a base rate. If the base rate indicator is B, the duty rate is a base rate; otherwise, space fill. <b>Not Used in RMS.</b>                                                                                                                 |
| o           | space fill                                | char(1)    | blank         | Space fill. <b>Not used in RMS.</b>                                                                                                                                                                                                                                               |
| V2 a        | Control identifier                        | char(1)    | V             | Identifies start of record                                                                                                                                                                                                                                                        |
| b           | Record type                               | char(1)    | 2             | Identifies record type                                                                                                                                                                                                                                                            |

| Record Name | Field Name                             | Field Type  | Default Value | Description                                                                                                                                                                                                                                                                                                                                                                  |
|-------------|----------------------------------------|-------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| c           | tariff number                          | Number (10) |               | A code located in the <i>Harmonized Tariff Schedule of the United States Annotated</i> (HTS) representing the tariff number. If this number is less than 10 positions, it is left justified. This number is the same as that in Record Identifier V1.                                                                                                                        |
| d           | general column 1 ad valorem percentage | Number (12) |               | The ad valorem rate of duty that appears in the General column of the HTS. Eight decimal places are implied.                                                                                                                                                                                                                                                                 |
| e           | column 1 other                         | Number (12) |               | The rate of duty that appears in the General column of the HTS that is not an ad valorem rate. Eight decimal places are implied.                                                                                                                                                                                                                                             |
| f           | Column 2 specific rate                 | Num(12)     |               | The specific rate of duty that appears in Column 2 of the HTS. Eight decimal places are implied.                                                                                                                                                                                                                                                                             |
| g           | Column 2 ad valorem percentage         | Num(12)     |               | The ad valorem rate of duty that appears in Column 2 of the HTS. Eight decimal places are implied.                                                                                                                                                                                                                                                                           |
| h           | Column 2 other rate                    | Num(12)     |               | The rate of duty that appears in Column 2 of the HTS that is not an ad valorem rate or a specific rate. Eight decimal places are implied.                                                                                                                                                                                                                                    |
| i           | countervailing duty flag               | char(1)     | blank or 1    | A code of 1 indicating the tariff number is subject to countervailing duty; otherwise, space fill.                                                                                                                                                                                                                                                                           |
| j           | additional tariff indicator            | char(1)     | blank or 'R'  | A code indicating if an additional tariff number may be required with this tariff number. Refer to the <i>Harmonized Tariff Schedule of the United States Annotated</i> (HTS) for more specific information on which HTS numbers require additional HTS numbers to be reported. This indicator is R when an additional tariff number may be required; otherwise, space fill. |
| k           | Miscellaneous Permit/License Indicator | char(2)     |               | A code indicating if a tariff number may be subject to a miscellaneous permit/license number.                                                                                                                                                                                                                                                                                |

| Record Name | Field Name             | Field Type | Default Value | Description                                                                                                                                                                                                                                                                                                                                                                            |
|-------------|------------------------|------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| l           | space fill             | char(4)    | blanks        | <b>Not used in RMS.</b>                                                                                                                                                                                                                                                                                                                                                                |
| V3     a    | Control identifier     | char(1)    | V             | identifies start of record                                                                                                                                                                                                                                                                                                                                                             |
| b           | Record type            | char(1)    | 3             | identifies record type                                                                                                                                                                                                                                                                                                                                                                 |
| c           | tariff number          | Number(10) |               | A code located in the <i>Harmonized Tariff Schedule of the United States Annotated</i> (HTS) representing the tariff number. If this number is less than 10 positions, it is left justified. This number is the same as the number in Record Identifier V1.                                                                                                                            |
| d           | GSP excluded countries | char(20)   |               | The International Organization for Standardization (ISO) country code that indicates countries not eligible for preferential treatment under GSP. Upto ten 2-position country codes can be reported. If countries are excluded from GSP, the Special Programs Indicator (SPI) Code contained in this record (positions 53-64) is A*. Valid ISO country codes are listed in Appendix B. |
| e           | OGA codes              | char(15)   |               | Codes that indicate special requirements by other Federal Government agencies must or may apply. Upto five 3-position OGA codes can be provided.                                                                                                                                                                                                                                       |
| f           | anti-dumping flag      | char(1)    | 1 or blank    | A code of 1 indicating the tariff number is subject to an antidumping duty; otherwise, space fill.                                                                                                                                                                                                                                                                                     |
| g           | quota indicator        | char(1)    | 1 or blank    | A code of 1 indicating the tariff number may be subject to quota. If the tariff number is not subject to quota, space fill.                                                                                                                                                                                                                                                            |
| h           | category number        | char(6)    |               | A code located in the HTS indicating the textile category assigned to the tariff number. If there is no textile category number, space fill.                                                                                                                                                                                                                                           |

| Record Name | Field Name                 | Field Type  | Default Value | Description                                                                                                                                                                                                                                                                              |
|-------------|----------------------------|-------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| I           | special program indicators | char(28)    |               | A code indicating if a tariff number is subject to a special program. Up to fourteen 2-position codes can be reported. Left justify. The SPI codes are not reported in any particular sequence. If more than fourteen 2-position codes are required, they are reported on the VD record. |
| NEWLINE     |                            |             | \n            |                                                                                                                                                                                                                                                                                          |
| <b>V4</b> a | Control identifier         | char(1)     | V             | identifies start of record. <b>Entire V4 record not used in RMS.</b>                                                                                                                                                                                                                     |
| b           | Record type                | char(1)     | 4             | identifies record type                                                                                                                                                                                                                                                                   |
| c           | tariff number              | Number (10) |               | A code located in the <i>Harmonized Tariff Schedule of the United States Annotated</i> (HTS) representing the tariff number. If this number is less than 10 positions, it is left justified. This number is the same as the number reported in Record Identifier V1.                     |
| d           | value edit code            | char(3)     |               | A code representing the value edit.                                                                                                                                                                                                                                                      |
| e           | value low bounds           | Number (10) |               | A value representing the minimum value edit. Five decimal places are implied. If this record contains date edits (positions 36-53), space fill.                                                                                                                                          |
| f           | value high bounds          | Number (10) |               | A value representing the maximum value edit. Five decimal places are implied. If this record contains date edits (positions 36-53), space fill.                                                                                                                                          |
| g           | entry date restriction     | Number (1)  | 0,1, or 2     | A code representing the first entry date restriction code.                                                                                                                                                                                                                               |
| h           | beginning restriction date | char(4)     |               | A numeric date in MMDD (month and day) format representing the first begin restriction date used in the edit. If this record contains a value edit (positions 13-35), space fill.                                                                                                        |



| Record Name | Field Name                   | Field Type  | Default Value    | Description                                                                                                                                                                                                                                                          |
|-------------|------------------------------|-------------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| I           | end restriction date         | char(4)     |                  | A numeric date in MMDD (month and day) format representing the first end restriction date used in the edit. If this record contains a value edit (positions 13-35), space fill.                                                                                      |
| j           | entry date restriction 2     | number(1)   | 0,1, or 2        | A code representing the second entry date restriction code.                                                                                                                                                                                                          |
| k           | beginning restriction date 2 | char(4)     |                  | A numeric date in MMDD (month and day) format representing the second begin restriction date used in the edit. If this record contains a value edit (positions 13-35), space fill.                                                                                   |
| l           | end restriction date 2       | char(4)     |                  | A code located in the <i>Harmonized Tariff Schedule of the United States Annotated</i> (HTS) representing the tariff number. If this number is less than 10 positions, it is left justified. This number is the same as the number reported in Record Identifier V1. |
| m           | country of origin            | char(2)     |                  | A code representing the value edit.                                                                                                                                                                                                                                  |
| n           | space filler                 | char(2)     | blanks           | A value representing the minimum value edit. Five decimal places are implied. If this record contains date edits (positions 36-53), space fill.                                                                                                                      |
| o           | quantity edit code           | char(3)     |                  | A value representing the maximum value edit. Five decimal places are implied. If this record contains date edits (positions 36-53), space fill.                                                                                                                      |
| p           | low quantity                 | Number (10) |                  | A code representing the first entry date restriction code.                                                                                                                                                                                                           |
| q           | high quantity                | Number (10) |                  | A numeric date in MMDD (month and day) format representing the first begin restriction date used in the edit. If this record contains a value edit (positions 13-35), space fill.                                                                                    |
| V5 a        | Control identifier           | char(1)     | V                | Identifies start of record                                                                                                                                                                                                                                           |
| b           | Record type                  | char(1)     | 5,6,7,8,9 ,A,B,C | Identifies record type                                                                                                                                                                                                                                               |

| Record Name | Field Name         | Field Type  | Default Value | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-------------|--------------------|-------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| c           | tariff number      | Number (10) |               | A code located in the <i>Harmonized Tariff Schedule of the United States Annotated</i> (HTS) representing the tariff number. If this number contains less than 10 positions, it is left justified. This number is the same as the number reported in Record Identifier V1.                                                                                                                                                                                                            |
| d           | Country code       | char(2)     |               | A code representing the country. Valid ISO country codes are listed in Appendix B. <i>E</i> followed by a space (Caribbean Basin Initiative), and <i>J</i> followed by a space (Andian Trade Preference Act), and <i>R</i> followed by a space (Caribbean Trade Partnership Act), are also valid codes for special rates. Countries eligible for E and J are indicated in the ACS country code file and the <i>Harmonized Tariff Schedule of the United States - Annotated</i> (HTS). |
| e           | specific rate      | Number (12) |               | The specific rate of duty listed in the Special column of the HTS. Eight decimal places are implied.                                                                                                                                                                                                                                                                                                                                                                                  |
| f           | ad valorem rate    | Number (12) |               | The ad valorem rate of duty listed in the Special column of the HTS. Eight decimal places are implied.                                                                                                                                                                                                                                                                                                                                                                                |
| g           | Other rate         | Number (12) |               | The rate of duty listed in the Special column of the HTS that is not a specific or ad valorem rate. Eight decimal places are implied.                                                                                                                                                                                                                                                                                                                                                 |
| h           | tax/fee class code | char(3)     |               | A code representing the tax/fee class. Valid tax/fee class codes are listed in Appendix B.                                                                                                                                                                                                                                                                                                                                                                                            |
| I           | tax/fee comp code  | char(1)     |               | A code indicating the first tax/fee computation formula. Computation formulas are presented in Appendix F.                                                                                                                                                                                                                                                                                                                                                                            |

| Record Name                                                                           | Field Name                           | Field Type  | Default Value     | Description                                                                                                                                                                                              |
|---------------------------------------------------------------------------------------|--------------------------------------|-------------|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| j                                                                                     | tax/fee flag                         | number(1)   |                   | A code indicating a tax/fee is required. Valid Tax/Fee Flag Codes are:<br><br>1 = Tax/fee required<br>2 = Tax/fee may be required. <b>Not used in RMS.</b>                                               |
| k                                                                                     | tax/fee specific rate                | Number (12) | blank if no value | The specific rate of duty required to compute taxes and/or fees. Eight decimal places are implied.                                                                                                       |
| l                                                                                     | tax/fee ad valorem                   | Number (12) | blank if no value | The ad valorem rate of duty required to compute taxes and/or fees. Eight decimal places are implied.                                                                                                     |
| m                                                                                     | space fill                           | char(1)     | blank             | Space fill.                                                                                                                                                                                              |
| <hr/> <b>Note: V6 through VC records have the same fields as the V5 record.</b> <hr/> |                                      |             |                   |                                                                                                                                                                                                          |
| <b>VD</b> a                                                                           | Control identifier                   | char(1)     | V                 | identifies start of record                                                                                                                                                                               |
| b                                                                                     | Record type                          | char(1)     | D                 | identifies record type                                                                                                                                                                                   |
| c                                                                                     | tariff number                        | Number (10) |                   | unique tariff number                                                                                                                                                                                     |
| d                                                                                     | Special Program Indicator (SPI) Code | char(32)    |                   | A code indicating if a tariff number is subject to a special program. Up to sixteen additional 2-position codes can be reported. Left justify. The SPI codes are not reported in any particular sequence |
| e                                                                                     | Filler                               | char(36)    |                   | Space fill.                                                                                                                                                                                              |

## Icadnld (Letter of Credit Application Download)

### Functional Area

Oracle Retail Trade Management - Letter of Credit

### Module Affected

LCADNLD.PC

## Design Overview

The LCADNLD program is used to process letter of credit (LC) applications, in order to be sent to the bank. Processing is based on the letter of credits flagged during the day by the LC applications action on the LC Find form, which is written to the staging table LC\_DOWNLOAD.

## Scheduling Constraints

| Schedule Information      | Description                                 |
|---------------------------|---------------------------------------------|
| Processing Cycle          | Phase 4 (daily)                             |
| Scheduling Considerations | Run LCADNLD before the LCMT700 Perl script. |
| Pre-processing            | N/A                                         |
| Post-Processing           | LCMT700 Perl script                         |
| Threading Scheme          | No threading due to low volume.             |

## Restart/Recovery

Restart/recovery for this program is set up at the lc\_ref\_id level. The recommended commit counter setting is 10000 records (subject to change based on experimentation).

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| LC_HEAD         | Yes    | No     | Yes    | No     |
| LC_DETAIL       | Yes    | No     | No     | No     |
| LC_DOWNLOAD     | Yes    | No     | No     | Yes    |
| OUTLOC          | Yes    | No     | No     | No     |
| ADDR            | Yes    | No     | No     | No     |
| SUP_IMPORT_ATTR | Yes    | No     | No     | No     |
| SUPS            | Yes    | No     | No     | No     |
| PARTNER         | Yes    | No     | No     | No     |
| ITEM_MASTER     | Yes    | No     | No     | No     |
| DOC             | Yes    | No     | No     | No     |

| Table   | Select | Insert | Update | Delete |
|---------|--------|--------|--------|--------|
| REQ_DOC | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name                     | Field Type | Default Value        | Description                                                                                                                        |
|-------------|--------------------------------|------------|----------------------|------------------------------------------------------------------------------------------------------------------------------------|
| File Header | File Type Record Descriptor    | Char(5)    | FHEAD                | Identifies file record type                                                                                                        |
|             | File Line Identifier           | Number(10) | line number in file  | ID of current line being created for output file.                                                                                  |
|             | File Type Definition           | Char(4)    | LCAP                 | Identifies file as 'Letter of Credit Application'                                                                                  |
|             | File Create Date               | Char(14)   | create date          | Current date, formatted to 'YYYYMMDDHH24MISS',                                                                                     |
| File Detail | File Type Record Descriptor    | Char(5)    | THEAD                | Identifies file record type                                                                                                        |
|             | File Line Sequence Number      | Number(10) | line number in file  | ID of current line being created for output file.                                                                                  |
|             | Transaction Set Control Number | Number(10) | sequence number      | Used to force unique file check                                                                                                    |
|             | Issuing Bank                   | Char(10)   | lc_head.issuing_bank | Used to sort the LCs into individualized bank SWIFT formatted files (using another program) – bank where LC application is headed. |
|             | Issuing Bank Name              | Char(240)  | partner.partner_desc | The description from the partner table where partner_id = issuing_bank and partner_type = 'BK'.                                    |
|             | Issuing Bank Address 1         | Char(240)  | addr.addr_1          | Mandatory line of address.                                                                                                         |

| Record Name | Field Name              | Field Type | Default Value         | Description                                                                                                                        |
|-------------|-------------------------|------------|-----------------------|------------------------------------------------------------------------------------------------------------------------------------|
|             | Issuing Bank Address 2  | Char(240)  | addr.add_2            | Non-mandatory line of address (can be null).                                                                                       |
|             | Issuing Bank Address 3  | Char(240)  | addr.add_3            | Non-mandatory line of address (can be null).                                                                                       |
|             | Issuing Bank City       | Char(120)  | addr.city             | City bank located in                                                                                                               |
|             | Issuing Bank State      | Char(3)    | addr.state            | State, if applicable, where bank located in.                                                                                       |
|             | Issuing Bank Post Code  | Char(30)   | addr.post             | Post code, if applicable, where bank located in.                                                                                   |
|             | Issuing Bank Country    | Char(3)    | addr.country_id       | Country bank located in.                                                                                                           |
|             | Advising Bank           | Char(10)   | lc_head.advising_bank | Used to sort the LCs into individualized bank SWIFT formatted files (using another program) – bank where LC application is headed. |
|             | Advising Bank Name      | Char(240)  | Partner.partner_desc  | The description from the partner table where partner_id = advising_bank and partner_type = 'BK'.                                   |
|             | Advising Bank Address 1 | Char(240)  | Addr.add_1            | Mandatory line of address.                                                                                                         |
|             | Advising Bank Address 2 | Char(240)  | Addr.add_2            | Non-mandatory line of address (can be null).                                                                                       |
|             | Advising Bank Address 3 | Char(240)  | Addr.add_3            | Non-mandatory line of address (can be null).                                                                                       |
|             | Advising Bank City      | Char(120)  | Addr.city             | City bank located in                                                                                                               |
|             | Advising Bank State     | Char(3)    | Addr.state            | State, if applicable, where bank located in.                                                                                       |

| Record Name | Field Name                        | Field Type | Default Value                 | Description                                                              |
|-------------|-----------------------------------|------------|-------------------------------|--------------------------------------------------------------------------|
|             | Advising Bank Post Code           | Char(30)   | Addr.post                     | Post code, if applicable, where bank located in.                         |
|             | Advising Bank Country             | Char(3)    | Addr.country_id               | Country bank located in.                                                 |
|             | Letter of Credit                  | Number(8)  | lc_head.lc_ref_id             | The LC_REF_ID off the LC_HEAD table.                                     |
|             | Form Type                         | Char(6)    | lc_head.form_type             | The level of detail that the LC sends to the issuing bank.               |
|             | Form Type Description             | Char(40)   | code_detail.code_desc         | Describes the form type: Long or Short.                                  |
|             | Letter of Credit Type             | Char(6)    | lc_head.lc_type               | The type of LC that is being applied for.                                |
|             | Letter of Credit Type Description | Char(40)   | code_detail.code_desc         | Describes the LC type: Master, Normal, Revolving.                        |
|             | Form of Letter of Credit – I      | Char(1)    | sup_import_attr.revocable_ind | The REVOCABLE_IND from the SUP_IMPORT_ATTR table.                        |
|             | Form of Letter of Credit – II     | Char(1)    | lc_head.transferable_ind      | Indicates if LC transferable.                                            |
|             | Application Date                  | Char(14)   | lc_head.application_date      | Date the LC is created within RTM/RMS, formatted to 'YYYYMMDD HH24MISS'. |
|             | Expiration Date                   | Char(14)   | lc_head.expiration_date       | The date the LC expires, formatted to 'YYYYMMDD HH24MISS'.               |
|             | Place of Expiry                   | Char(6)    | lc_head.place_of_expiry       | Code for the place the LC expires.                                       |

| Record Name | Field Name                  | Field Type | Default Value                      | Description                                                                                  |
|-------------|-----------------------------|------------|------------------------------------|----------------------------------------------------------------------------------------------|
|             | Place of Expiry Description | Char(40)   | desc is retrieved through a decode | The description of the place the LC expires.                                                 |
|             | Applicant                   | Char(10)   | lc_head.a<br>pplicant              | Party on whose behalf the LC is being issued.                                                |
|             | Applicant Name              | Char(240)  | partner.pa<br>rtner_des<br>c       | The description from the partner table where partner_id = applicant and partner_type = 'AP'. |
|             | Applicant Address 1         | Char(240)  | addr.add_<br>1                     | Mandatory line of address.                                                                   |
|             | Applicant Address 2         | Char(240)  | addr.add_<br>2                     | Non-mandatory line of address (can be null).                                                 |
|             | Applicant Address 3         | Char(240)  | addr.add_<br>3                     | Non-mandatory line of address (can be null).                                                 |
|             | Applicant City              | Char(120)  | addr.city                          | City applicant located in                                                                    |
|             | Applicant State             | Char(3)    | addr.state                         | State, if applicable, where applicant located in.                                            |
|             | Applicant Post Code         | Char(10)   | addr.post                          | Post code, if applicable, where applicant located in.                                        |
|             | Applicant Country           | Char(3)    | addr.coun<br>try_id                | Country applicant located in.                                                                |
|             | Beneficiary                 | Number(10) | lc.head.be<br>neficiary            | Party in favor of which the LC is being issued.                                              |
|             | Beneficiary Name            | Char(240)  | sup.s.<br>sup_name                 | Beneficiary (supplier) name from the SUPS table,.                                            |
|             | Beneficiary Address 1       | Char(240)  | addr.add_<br>1                     | Mandatory line of address.                                                                   |
|             | Beneficiary Address 2       | Char(240)  | addr.add_<br>2                     | Non-mandatory line of address (can be null).                                                 |
|             | Beneficiary Address 3       | Char(240)  | addr.add_<br>3                     | Non-mandatory line of address (can be null).                                                 |
|             | Beneficiary City            | Char(120)  | addr.city                          | City beneficiary located in                                                                  |
|             | Beneficiary State           | Char(3)    | addr.state                         | State, if applicable, where beneficiary located in.                                          |



| Record Name | Field Name                     | Field Type     | Default Value                      | Description                                                  |
|-------------|--------------------------------|----------------|------------------------------------|--------------------------------------------------------------|
|             | Beneficiary Post Code          | Char(30)       | addr.post                          | Post code, if applicable, where beneficiary located in.      |
|             | Beneficiary Country            | Char(3)        | addr.country_id                    | Country beneficiary located in.                              |
|             | Currency Code                  | Char(3)        | lc_head.currency_code              | The country of origin for the orders on the LC.              |
|             | Exchange Rate                  | Number (20,10) | lc_head.exchange_rate              | Exchange_rate to convert LC currency to RMS currency.        |
|             | Origin Country ID              | Char(3)        | lc_head.origin_country_id          | Origin country of the orders associated with the LC.         |
|             | Presentation Terms             | Char(6)        | lc_head.presentation_terms         | Code for the terms of presentation.                          |
|             | Presentation Terms Description | Char(40)       | desc is retrieved through a decode | Description of the terms of presentation.                    |
|             | Purchase Type                  | Char(6)        | lc_head.purchase_type              | Code for the purchase type.                                  |
|             | Purchase Type Description      | Char(40)       | desc is retrieved through a decode | Description of the purchase type.                            |
|             | Advice Method                  | Char(6)        | lc_head.advice_method              | Code for the advice method.                                  |
|             | Advice Method Description      | Char(40)       | desc is retrieved through a decode | Description of the advice method (eg. Full Wire, Mail, etc). |
|             | Issuance                       | Char(6)        | lc_head.issuance                   | Code for the issuance.                                       |
|             | Issuance Description           | Char(40)       | desc is retrieved through a decode | Description of the issuance (eg. Cable, Telex, etc).         |

| Record Name | Field Name                | Field Type    | Default Value                      | Description                                                                                                 |
|-------------|---------------------------|---------------|------------------------------------|-------------------------------------------------------------------------------------------------------------|
|             | Amount Type               | Char(6)       | lc_head.amount_type                | If 'E'xact, then amount must be exact, if 'A'pproximate then amount can be within variance percent.         |
|             | Amount Type Description   | Char(40)      | desc is retrieved through a decode | Description of amount_type                                                                                  |
|             | Amount                    | Number (20,4) | lc_head.amount                     | The total amt of the Letter of Credit                                                                       |
|             | Variance Percent          | Number (12,4) | lc_head.variance_percent           | Allowed currency variance percent for the LC                                                                |
|             | Specification             | Char(6)       | lc_head.specification              | Code for any condition for the credit, that is, "maximum", etc.                                             |
|             | Specification Description | Char(40)      | desc is retrieved through a decode | Description of condition for the credit, that is, "maximum", etc.                                           |
|             | Credit Available With     | Char(10)      | lc_head.credit_availability_with   | Code for bank with which credit is available.                                                               |
|             | Credit With Bank Name     | Char(40)      | partner.partner_description        | The description from the partner table where partner_id = credit_availability_with and partner_type = 'BK'. |
|             | Credit With Address 1     | Char(240)     | addr.address_1                     | Mandatory line of address.                                                                                  |
|             | Credit With Address 2     | Char(240)     | addr.address_2                     | Non-mandatory line of address (can be null).                                                                |
|             | Credit With Address 3     | Char(240)     | addr.address_3                     | Non-mandatory line of address (can be null).                                                                |
|             | Credit With City          | Char(120)     | addr.city                          | City creditor located in                                                                                    |
|             | Credit With State         | Char(3)       | addr.state                         | State, if applicable, where creditor located in.                                                            |
|             | Credit With Post Code     | Char(30)      | addr.post                          | Post code, if applicable, where creditor located in.                                                        |

| Record Name | Field Name                 | Field Type | Default Value                      | Description                                                                                         |
|-------------|----------------------------|------------|------------------------------------|-----------------------------------------------------------------------------------------------------|
|             | Credit With Country        | Char(3)    | addr.country_id                    | Country creditor located in.                                                                        |
|             | Drafts At                  | Char(6)    | lc_head.drafts_at                  | Specifies the terms of the drafts to be drawn under the LC.                                         |
|             | Drafts At Description      | Char(40)   | desc is retrieved through a decode | Description of the terms of the drafts to be drawn under the LC.                                    |
|             | Drawee                     | Char(10)   | lc_head.paying_bank                | Identifies drawee of drafts to be drawn under LC (paying bank)                                      |
|             | Drawee Name                | Char(240)  | partner.partner_desc               | The description from the partner table where partner_id = paying_bank and partner_type = 'BK'.      |
|             | Drawee Address 1           | Char(240)  | addr.addr_1                        | Mandatory line of address.                                                                          |
|             | Drawee Address 2           | Char(240)  | addr.addr_2                        | Non-mandatory line of address (can be null).                                                        |
|             | Drawee Address 3           | Char(240)  | addr.addr_3                        | Non-mandatory line of address (can be null).                                                        |
|             | Drawee City                | Char(120)  | addr.city                          | City bank located in                                                                                |
|             | Drawee State               | Char(3)    | addr.state                         | State, if applicable, where bank located in.                                                        |
|             | Drawee Post Code           | Char(30)   | addr.post                          | Post code, if applicable, where bank located in.                                                    |
|             | Drawee Country             | Char(3)    | addr.country_id                    | Country bank located in.                                                                            |
|             | Negotiating Bank           | Char(10)   | lc_head.negotiating_bank           | Identifies the negotiating bank.                                                                    |
|             | Negotiating Bank Name      | Char(240)  | partner.partner_desc               | The description from the partner table where partner_id = negotiating_bank and partner_type = 'BK'. |
|             | Negotiating Bank Address 1 | Char(240)  | addr.addr_1                        | Mandatory line of address.                                                                          |

| Record Name | Field Name                 | Field Type | Default Value           | Description                                                                                        |
|-------------|----------------------------|------------|-------------------------|----------------------------------------------------------------------------------------------------|
|             | Negotiating Bank Address 2 | Char(240)  | addr.add_2              | Non-mandatory line of address (can be null).                                                       |
|             | Negotiating Bank Address 3 | Char(240)  | addr.add_3              | Non-mandatory line of address (can be null).                                                       |
|             | Negotiating Bank City      | Char(120)  | addr.city               | City bank located in                                                                               |
|             | Negotiating Bank State     | Char(3)    | addr.state              | State, if applicable, where bank located in.                                                       |
|             | Negotiating Bank Post Code | Char(30)   | addr.post               | Post code, if applicable, where bank located in.                                                   |
|             | Negotiating Bank Country   | Char(3)    | addr.country_id         | Country bank located in.                                                                           |
|             | Confirming Bank            | Char(10)   | lc_head.confirming_bank | Identifies the confirming bank.                                                                    |
|             | Confirming Bank Name       | Char(240)  | partner.partner_desc    | The description from the partner table where partner_id = confirming_bank and partner_type = 'BK'. |
|             | Confirming Bank Address 1  | Char(240)  | addr.add_1              | Mandatory line of address.                                                                         |
|             | Confirming Bank Address 2  | Char(240)  | addr.add_2              | Non-mandatory line of address (can be null).                                                       |
|             | Confirming Bank Address 3  | Char(240)  | addr.add_3              | Non-mandatory line of address (can be null).                                                       |
|             | Confirming Bank City       | Char(120)  | addr.city               | City bank located in                                                                               |
|             | Confirming Bank State      | Char(3)    | addr.state              | State, if applicable, where bank located in.                                                       |
|             | Confirming Bank Post Code  | Char(30)   | addr.post               | Post code, if applicable, where bank located in.                                                   |

| Record Name | Field Name                  | Field Type | Default Value                     | Description                                                                                          |
|-------------|-----------------------------|------------|-----------------------------------|------------------------------------------------------------------------------------------------------|
|             | Confirming Bank Country     | Char(3)    | addr.coun<br>try_id               | Country bank located in.                                                                             |
|             | Transferring Bank           | Char(10)   | lc_head.tr<br>ansferring<br>_bank | Identifies the transferring bank.                                                                    |
|             | Transferring Bank Name      | Char(240)  | partner.pa<br>rtner_des<br>c      | The description from the partner table where partner_id = transferring_bank and partner_type = 'BK'. |
|             | Transferring Bank Address 1 | Char(240)  | addr.add_<br>1                    | Mandatory line of address.                                                                           |
|             | Transferring Bank Address 2 | Char(240)  | addr.add_<br>2                    | Non-mandatory line of address (can be null).                                                         |
|             | Transferring Bank Address 3 | Char(240)  | addr.add_<br>3                    | Non-mandatory line of address (can be null).                                                         |
|             | Transferring Bank City      | Char(120)  | addr.city                         | City bank located in                                                                                 |
|             | Transferring Bank State     | Char(3)    | addr.state                        | State, if applicable, where bank located in.                                                         |
|             | Transferring Bank Post Code | Char(30)   | addr.post                         | Post code, if applicable, where bank located in.                                                     |
|             | Transferring Bank Country   | Char(3)    | addr.coun<br>try_id               | Country bank located in.                                                                             |
|             | Partial Shipment Indicator  | Char(1)    | lc_head.p<br>artial_shi<br>p_ind  | Indicates whether goods covered by LC can be partially shipped or not.                               |
|             | Transshipment Indicator     | Char(1)    | lc_head.tr<br>ansshipm<br>ent_ind | Indicates whether goods can be transferred to another vessel midway through the voyage.              |
|             | Fob Title Pass              | Char(6)    | lc_head.f<br>ob_title_p<br>ass    | Indicates where the title for goods is passed from the vendor to the purchaser.                      |

| Record Name | Field Name                    | Field Type | Default Value                      | Description                                                                         |
|-------------|-------------------------------|------------|------------------------------------|-------------------------------------------------------------------------------------|
|             | Fob Title Pass Decode         | Char(40)   | desc is retrieved through a decode | Decode of where the title for goods is passed from the vendor to the purchaser.     |
|             | Fob Title Pass Description    | Char(250)  | lc_head.ob_title_pass_desc         | Describes the FOB_TITLE_PASS – could be city name etc.                              |
|             | Transportation to             | Char(5)    | lc_head.transportation_to          | Transportation to location                                                          |
|             | transportation to description | Char(150)  | outloc.outloc_desc                 | Description of transportation to location                                           |
|             | With Recourse Indicator       | Char(1)    | lc_head.with_recourse_ind          | Indicates conditional payment on the part of the bank as instructed by the buyer.   |
|             | Latest Shipment Date          | Char(14)   | lc_head.latest_ship_date           | Latest ship date for all Pos included in the LC, formatted to 'YYYYMMDD HH24MISS'   |
|             | Earliest Shipment Date        | Char(14)   | lc_head.earliest_ship_date         | Earliest ship date for all Pos included in the LC, formatted to 'YYYYMMDD HH24MISS' |

| Record Name | Field Name                        | Field Type                                                                                                                                                            | Default Value       | Description                                                         |
|-------------|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|---------------------------------------------------------------------|
|             | Letter of Credit Negotiation Days | Number(3)<br>replaces x<br>in the string<br>“DOCUMENTS TO BE PRESENTED WITHIN x DAYS AFTER ISSUANCE OF THE SHIPPING DOCUMENTS BUT WITHIN THE VALIDITY OF THIS CREDIT” | lc.head.lc_neg_days | The number of days to negotiate documents                           |
|             | Bank's LC reference id            | Number(8)                                                                                                                                                             | lc_head.bank_lc_id  | Bank's LC ref id                                                    |
|             | File Type Record Descriptor       | Char(5)                                                                                                                                                               | THDCM               | Identifies file record type                                         |
|             | File Line Sequence Number         | Number(10)                                                                                                                                                            | line number in file | ID of current line being created for output file                    |
|             | Transaction Set Control Number    | Number(10)                                                                                                                                                            | sequence number     | Used to force unique file check                                     |
|             | Header Level Comments             | Char(2000)                                                                                                                                                            | lc_head.comments    | Holds any comments that the user has added to the Letter of Credit. |
|             | File Type Record Descriptor       | Char(5)                                                                                                                                                               | TDOCS               | Identifies file record type                                         |
|             | File Line Sequence Number         | Number(10)                                                                                                                                                            | line number in file | ID of current line being created for output file                    |
|             | Transaction Set Control Number    | Number(10)                                                                                                                                                            | sequence number     | Used to force unique file check                                     |

| Record Name | Field Name                     | Field Type    | Default Value                | Description                                                                                                                            |
|-------------|--------------------------------|---------------|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
|             | Swift Tag                      | Char(6)       | doc.swift_tag                | Identifies individual document types that can be associated with an LC.                                                                |
|             | Document ID                    | Number(6)     | req_doc.doc_id               | Uniquely identifies the individual documents associated with an LC.                                                                    |
|             | Body Text                      | Char(2000)    | req_doc.doc_text             | Documents associated with a given LC.<br>Description of Goods and Services OR Documents Required OR Additional Conditions OR Narrative |
|             | File Type Record Descriptor    | Char(5)       | TDETL                        | Identifies file record type                                                                                                            |
|             | File Line Sequence Number      | Number(10)    | line number in file          | ID of current line being created for output file                                                                                       |
|             | Transaction Set Control Number | Number(10)    | sequence number              | Used to force unique file check                                                                                                        |
|             | Order Number                   | Number(8)     | lc_detail.order_no           | PO associated with the LC.                                                                                                             |
|             | Item                           | Char(25)      | lc_detail.item               | Item on the PO – item is rolled up to the item_level of 1, if possible.                                                                |
|             | Cost                           | Number (20,4) | lc_detail.cost               | If form_type = 'S'hort then cost is the total cost of the order; if the form_type = 'L'ong then the cost is the unit cost of the item. |
|             | Quantity                       | Number (12,4) | lc_detail.qty                | Total qty of the item for the order on the LC.                                                                                         |
|             | Standard UOM                   | Char(4)       | Item_master.standard_uom     | Standard unit of measure of the quantity of the item for the order on the LC.                                                          |
|             | Earliest Ship Date             | Char(14)      | lc_detail.earliest_ship_date | The earliest date an order on the LC can be shipped, formatted to 'YYYYMMDDHH24MISS',                                                  |



| Record Name  | Field Name                     | Field Type | Default Value              | Description                                                                         |
|--------------|--------------------------------|------------|----------------------------|-------------------------------------------------------------------------------------|
|              | Latest Ship Date               | Char(14)   | lc_detail.latest_ship_date | The latest date an order on the LC can be shipped, formatted to 'YYYYMMDD HH24MISS' |
|              | item description               | Char(250)  | Item_master.desc_up        | Item's description                                                                  |
|              | File Type Record Descriptor    | Char(5)    | TMERC                      | Identifies file record type                                                         |
|              | File Line Sequence Number      | Number(10) | line number in file        | ID of current line being created for output file                                    |
|              | Transaction Set Control Number | Number(10) | sequence number            | Used to force unique file check                                                     |
|              | Merchandise Description        | Char(2000) | lc_detail.merch_desc       | Contains the merchandise description of the field.                                  |
|              | File Type Record Descriptor    | Char(5)    | TDTCM                      | Identifies file record type                                                         |
|              | File Line Sequence Number      | Number(10) | line number in file        | ID of current line being created for output file                                    |
|              | Transaction Set Control Number | Number(10) | sequence number            | Used to force unique file check                                                     |
|              | Detail Level Comments          | Char(2000) | lc_detail.comments         | Holds any comments that the user has added to the Letter of Credit detail record.   |
| File Trailer | File Type Record Descriptor    | Char(5)    | TTAIL                      | Identifies file record type                                                         |
|              | File Line Sequence Number      | Number(10) | line number in file        | ID of current line being created for output file                                    |

| Record Name  | Field Name                     | Field Type | Default Value                                    | Description                                                                                 |
|--------------|--------------------------------|------------|--------------------------------------------------|---------------------------------------------------------------------------------------------|
|              | Transaction Set Control Number | Number(10) | sequence number                                  | Used to force unique file check                                                             |
|              | Transaction detail line count  | Number(10) | ID of current line being created for output file | Sum of the detail lines within a transaction                                                |
| File Trailer | File Type Record Descriptor    | Char(5)    | FTAIL                                            | Identifies file record type                                                                 |
|              | File Line Identifier           | Number(10) | Sequential number Created by program.            | ID of current line being created for output file.                                           |
|              | File Record Counter            | Number(10) |                                                  | Number of records/transactions processed in current file (only records between head & tail) |

## lcmdnld (Letter of Credit Amendment Download)

### Functional Area

Oracle Retail Trade Management - Letter of Credit

### Module Affected

LCMDNLD.PC

### Design Overview

LCMDNLD.PC downloads amended letter of credit information to a bank, in the S.W.I.F.T. format. Processing is based on the amendments flagged during the day by the Select LC Amendments action on the LC Find form (lcfind.fmb), which is then written to the staging table LC\_DOWNLOAD.

### Scheduling Constraints

| Schedule Information      | Description     |
|---------------------------|-----------------|
| Processing Cycle          | Phase 4 (Daily) |
| Scheduling Considerations | N/A             |

| Schedule Information | Description                     |
|----------------------|---------------------------------|
| Pre-Processing       | N/A                             |
| Post-Processing      | lcmt707 perl script             |
| Threading Scheme     | No threading due to low volume. |

### Restart/Recovery

Restart/recovery for this program is set up at the lc\_ref\_id level. The recommended commit counter setting is 1000 records (subject to change based on experimentation).

### Locking Strategy

N/A

### Security Considerations

N/A.

### Performance Considerations

N/A

### Key Tables Affected

| Table         | Select | Insert | Update | Delete |
|---------------|--------|--------|--------|--------|
| LC_AMENDMENTS | Yes    | No     | Yes    | No     |
| LC_HEAD       | Yes    | No     | No     | No     |
| LC_DOWNLOAD   | Yes    | No     | No     | Yes    |
| ADDR          | Yes    | No     | No     | No     |
| PARTNER       | Yes    | No     | No     | No     |
| SUPS          | Yes    | No     | No     | No     |
| CODE_DETAIL   | Yes    | No     | No     | No     |
| ITEM_MASTER   | Yes    | No     | No     | No     |
| DOC           | Yes    | No     | No     | No     |
| REQ_DOC       | Yes    | No     | No     | No     |

### I/O Specification

#### Output File Layout

| Record Name | Field Name                        | Field Type | Default Value | Description                 |
|-------------|-----------------------------------|------------|---------------|-----------------------------|
| File Header | File Type<br>Record<br>Descriptor | Char(5)    | FHEAD         | Identifies file record type |

| Record Name        | Field Name                     | Field Type  | Default Value        | Description                                                                                                                        |
|--------------------|--------------------------------|-------------|----------------------|------------------------------------------------------------------------------------------------------------------------------------|
|                    | File Line Sequence Number      | Number(10)  | Line number in file  | Keeps track of the record's position in the file by line number                                                                    |
|                    | File Type Definition           | Char(4)     | LCAM                 | Identifies file as 'Letter of Credit Amendment'                                                                                    |
|                    | File Create Date               | Char(14)    | Create date          | Current date, formatted to 'YYYYMMDDHH24MISS'                                                                                      |
| Transaction Header | Filetype Record descriptor     | Char(5)     | THEAD                | Identifies file record type                                                                                                        |
|                    | File Line Sequence Number      | Number (10) | Line number in file  | Keeps track of the record's position in the file by line number                                                                    |
|                    | Transaction Set Control Number | Number (10) | Sequence number      | Used to force unique file check                                                                                                    |
|                    | Issuing Bank                   | Char(10)    | lc_head.issuing_bank | Used to sort the LCs into individualized bank SWIFT formatted files (using another program) – bank where LC application is headed. |
|                    | Issuing Bank Name              | Char(240)   | partner.partner_desc | The description from the partner table where partner_id = issuing_bank and partner_type = 'BK'.                                    |
|                    | Issuing Bank Address 1         | Char(240)   | addr.add_1           | Mandatory line of address.                                                                                                         |
|                    | Issuing Bank Address 2         | Char(240)   | addr.add_2           | Non-mandatory line of address (can be null).                                                                                       |
|                    | Issuing Bank Address 3         | Char(240)   | addr.add_3           | Non-mandatory line of address (can be null).                                                                                       |
|                    | Issuing Bank City              | Char(120)   | addr.city            | City bank located in                                                                                                               |
|                    | Issuing Bank State             | Char(3)     | addr.state           | State, if applicable, where bank located in.                                                                                       |
|                    | Issuing Bank Post Code         | Char(30)    | addr.post            | Post code, if applicable, where bank located in.                                                                                   |
|                    | Issuing Bank Country           | Char(3)     | addr.country_id      | Country bank located in.                                                                                                           |

| Record Name        | Field Name                            | Field Type    | Default Value          | Description                                                                                                                                                                                                                              |
|--------------------|---------------------------------------|---------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                    | Letter of Credit                      | Number (8)    | lc_detail.lc_ref_id    | The LC_REF_ID off the LC_DETAIL table.                                                                                                                                                                                                   |
|                    | Bank Letter of Credit ID              | Char(16)      | lc_head.bank_lc_id     | The BANK_LC_ID off the LC_HEAD table.                                                                                                                                                                                                    |
|                    | Currency Code                         | Char(3)       | lc_head.currency_code  | The CURRENCY_CODE off the LC_HEAD table.                                                                                                                                                                                                 |
|                    | Date of Issue/ Transfer of the Credit | Char(14)      | lc_head.confirmed_date | Date the Issuing Bank thinks is the date of issue—when it was officially confirmed, formatted to 'YYYYMMDDHH24MISS'.                                                                                                                     |
|                    | Current Amount of LC                  | Number (20,4) |                        | This amount is calculated in the get_current_amount() function and is the net amount of the LC calculated only using amendments that are downloaded. Normally, the net amount is calculated using amendments in the 'D'ownloaded status. |
|                    | Beneficiary                           | Number (10)   | lc.head.beneficiary    | Party in favor of which the LC is being issued.                                                                                                                                                                                          |
|                    | Beneficiary Name                      | Char(240)     | supsup_name            | Beneficiary (supplier) name from the SUPS table.                                                                                                                                                                                         |
|                    | Beneficiary Address 1                 | Char(240)     | addr.add_1             | Mandatory line of address.                                                                                                                                                                                                               |
|                    | Beneficiary Address 2                 | Char(240)     | addr.add_2             | Non-mandatory line of address (can be null).                                                                                                                                                                                             |
|                    | Beneficiary Address 3                 | Char(240)     | addr.add_3             | Non-mandatory line of address (can be null).                                                                                                                                                                                             |
|                    | Beneficiary City                      | Char(120)     | addr.city              | City beneficiary located in                                                                                                                                                                                                              |
|                    | Beneficiary State                     | Char(3)       | addr.state             | State, if applicable, where beneficiary located in.                                                                                                                                                                                      |
| Transaction Detail | Beneficiary Post Code                 | Char(30)      | addr.post              | Post code, if applicable, where beneficiary located in.                                                                                                                                                                                  |
|                    | Beneficiary Country                   | Char(3)       | addr.country_id        | Country beneficiary located in.                                                                                                                                                                                                          |
|                    | File Type Record Descriptor           | Char(5)       | TDETL                  | Identifies file record type                                                                                                                                                                                                              |

| Record Name | Field Name                     | Field Type  | Default Value               | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------|--------------------------------|-------------|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | File Line Sequence Number      | Number (10) | line number in file         | Keeps track of the record's position in the file by line number                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|             | Transaction Set Control Number | Number (10) | sequence number             | Used to force unique file check                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|             | Amendment Number               | Number (8)  | lc_amendments.amend_no      | Holds the amendment number for the amendment.                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|             | Order_no                       | Number (8)  | lc_amendments.order_no      | Order_no, if applicable, that is attached to the LC that is being amended.                                                                                                                                                                                                                                                                                                                                                                                                                               |
|             | Item                           | Char(25)    | lc_amendments.item          | Item being amended, either a Style or Staple sku.                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|             | Value Being Amended            | Char(6)     | lc_amendments.amended_value | LC Field being amended. Can be any of the following code_types:<br>CODE CODE_DESC<br>-----<br>AI Add Item<br>AO Add PO<br>ARQD Add Req'd Doc.<br>C Cost<br>ED Expiration Date<br>ESD Earliest Ship Date<br>LSD Latest Ship Date<br>NA Net Amount<br>ND Negotiation Days<br>OC Origin Country<br>OQ Order Quantity<br>PE Place of Expiry<br>PRT Presentation Terms<br>PSF Partial Ship Flag<br>RI Remove Item<br>RO Remove PO<br>RRQD Remove Req'd Doc<br>TFF Transferable Flag<br>TSF Transshipment Flag |

| Record Name      | Field Name                      | Field Type    | Default Value                | Description                                                                                                                                                                                                               |
|------------------|---------------------------------|---------------|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                  | Value Being Amended Description | Char(40)      | code_detail.code_desc        | The Value Being Amended decoded (see the above list). May be used when printing to the SWIFT file MT 707 for clarity.                                                                                                     |
|                  | Original Value of Amended Field | Char(45)      | lc_amendments.original_value | Current value of field that is being amended.                                                                                                                                                                             |
|                  | New Value of Amended Field      | Char (2000)   | lc_amendments.new_value      | New value of the field that is being amended.                                                                                                                                                                             |
|                  | Description of New Value        | Char(40)      | code_detail.code_desc        | The new value decoded (or fetched from a table, as in the origin_country case)– only applicable to the following amended values: place of expiry, title_pass_location, origin_country, presentation terms, purchase type. |
|                  | Sign                            | Char(1)       |                              | If the effect is negative it is “-“ if the effect is positive it is “ “.                                                                                                                                                  |
|                  | Effect                          | Number (20,4) | lc.amendments.effect         | Effect that amendment has on LC if amendment to change qty or cost of a PO or amount of LC itself.                                                                                                                        |
|                  | Date of Amendment               | Char(14)      | Lc_amendments.accept_date    | Date on which Issuing Bank (or issuing party, in this case the retailer) considers the credit as being amended, formatted to ‘YYYYMMDD HH24MISS’.                                                                         |
| Transaction Text | File Type Record Descriptor     | Char(5)       | TTEXT                        | Identifies file record type                                                                                                                                                                                               |
|                  | File Line Sequence Number       | Number (10)   | line number in file          | Keeps track of the record’s position in the file by line number                                                                                                                                                           |
|                  | Transaction Set Control Number  | Number (10)   | sequence number              | Used to force unique file check                                                                                                                                                                                           |

| Record Name  | Field Name                     | Field Type  | Default Value       | Description                                                                                                                            |
|--------------|--------------------------------|-------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------|
|              | Amendment Text                 | Char (2000) | text description    | A text description of the individual amendment (for each TDETL line of the output file) built by the package LC_AMEND_SQL. AMEND_TEXT. |
| File Trailer | File Type Record Descriptor    | Char(5)     | FTAIL               | Identifies file record type                                                                                                            |
|              | File Line Sequence Number      | Number (10) | line number in file | Keeps track of the record's position in the file by line number                                                                        |
|              | Control Number File Line Count | Number (10) | total detail lines  | Sum of all transaction lines, not including the file header and trailer                                                                |

## lcmt700 (SWIFT File Conversion)

### Functional Area

Oracle Retail Trade Management – Letter of Credit Interfaces

### Module Affected

LCMT700

### Design Overview

This Perl script converts the standard RMS flat file into the bank specific S.W.I.F.T. MT 700 output files. The S.W.I.F.T file is in a different file layout than the standard Oracle Retail interface file format. The input file for this Perl script is the output of the lcadnld.pc RMS batch. One output file is created for each issuing bank in the interface file.

### Scheduling Constraints

| Schedule Information      | Description                                                                          |
|---------------------------|--------------------------------------------------------------------------------------|
| Processing Cycle          | N/A                                                                                  |
| Scheduling Considerations | LCMT700 should run after Letter of Credit application download program (LCADNLD.PC). |
| Pre-Processing            | N/A                                                                                  |
| Post-Processing           | N/A                                                                                  |
| Threading Scheme          | N/A                                                                                  |



**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| N/A   |        |        |        |        |

**I/O Specification**

The input file to this Perl script is the output from LCADNLD.PC. Please refer to the LCADNLD design for details on the input file layout.

- All files layouts input and output the SWIFT MT 700. The output file should be in the following format:
- Most output fields are contained in their own line (or 3-4 line for addresses).
- Each application consists of four parts, one MT 700 and three MT 701s, which are ordered through the Sequence of Total field: for example, ‘:27:1/4 MT 700’ is the first (MT 700) part of the application.
- MT 700 and MT 701s is mingled in the same file.
- Each record starts with a colon and a SWIFT field identifier, followed by another colon: for example, ‘:40A:’-
- Each application is separated by a line with only the ASCII 3 symbol (a heart) on it.

**Examples of how individual lines of the MT 700 or MT 701 should look:**

```
:27:1/4
:40A:IRREVOCABLE
:20:29893098
:23:NOREF
:31C:910906
:31D:911022DALLAS
:51D:NORTHERN TRUST INT’L BANKING CORP.
      ONE WORLD TRADE CENTER
SUITE 3941
NY, NY 10048 USA
```

**The layout of the S.W.I.F.T MT 700 (Issue of a Documentary Credit) file is as follows:**

SWIFT I.D. DATA TYPE CODES (refer to SWIFT User Handbook – Standards general Information – October 1998 release for formatting information):

**Notes:**

- There is always a new line (nl) after every individual SWIFT ID (and there may be more than one line within an individual field [for example, 59 – Beneficiary, four lines to hold address information]).
- In some situations, certain fields are blank. These fields should be skipped over. In other words, no blank line or tag should be printed indicating the field is blank. Simply ignore it.

## **lcmt707 (Converts Letter of Credit from Oracle Retail Format to SWIFT)**

**Functional Area**

Oracle Retail Trade Management – Letter of Credit Interfaces

**Module Affected**

LCMT707

**Design Overview**

This Perl script converts the Oracle retail standard interface file format for Amendments to Letters of Credit download to the corresponding S.W.I.F.T file format (MT 707). The input file for this Perl script is the output of the LCMDNLD.PC RMS batch.

**Scheduling Constraints**

| <b>Schedule Information</b> | <b>Description</b>                                                                 |
|-----------------------------|------------------------------------------------------------------------------------|
| Processing Cycle            | N/A                                                                                |
| Scheduling Considerations   | LCMT707 should run after Letter of Credit amendment download program (lcmdnld.pc). |
| Pre-Processing              | N/A                                                                                |
| Post-Processing             | N/A                                                                                |
| Threading Scheme            | N/A                                                                                |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

## Key Tables Affected

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| N/A   |        |        |        |        |

## I/O Specification

The input file to this Perl script is the output from LCMDNLD.PC. Please refer to the LCMDNLD design for details on the input file layout.

### The SWIFT MT 707 output file should be in the following format:

- Most output fields are contained in their own line (or 3-4 line for addresses).
- Each amendment consists of only one part, the MT 707. There may be several MT 707s at any given time associated to an LC because they are grouped by amendment number at the time of creation. All TDETL records with the same amend\_no is grouped together in one MT 707.
- Each record starts with a colon and a SWIFT field identifier, followed by another colon: for example, ':40A: '-
- Each amendment is separated by a line with only the ASCII 3 symbol (a heart) on it.

### Logic Setup:

The input file is in standard RMS file format. It potentially has numerous TDETL lines per each THEAD line. There may be numerous TDETL records for one amendment. MT 707 writes one record for each amendment, so if there are multiple TDETL records they need to be combined. There is one TTEXT for each TDETL.

There are three values that need to be calculated. 32B, 33B, 34B. 32B is the total increment or the sum of the positive effect values for each amendment. 33B is the total decrement or the sum of all the negative effect values for each amendment. 32B and 33B are separate totals for each amendment. 34B is the total difference, so it is the sum of the total increment and total decrement. 34B is not just for one amendment though; it is for all amendments of a THEAD record, so this total runs through each TDETL in a THEAD.

### For example: if the input file contains:

```
THEAD...
TDETL amendment 1, effect +1000
TTEXT
TDETL amendment 1, effect +500
TTEXT
TDETL amendment 2, effect -2500
TTEXT
TDETL amendment 3, effect +4000
TTEXT
TDETL amendment 3, effect -1000
TTEXT
TDETL amendment 3, effect +500
TTEXT
TDETL amendment 4, effect -1000
TTEXT
TDETL amendment 4 , effect -2500
TTEXT
TTAIL
```

32B for amendment 1 = 1500

33B for amendment 1 = 0

34B for amendemnt 1 = 1500

32B for amendment 2 = 0  
33B for amendment 2 = 2500  
34B for amendemnt 2 = -1000

32B for amendment 3 = 4500  
33B for amendment 3 = 1000  
34B for amendemnt 3 = 4500

32B for amendment 4 = 0  
33B for amendment 4 = 3500  
34B for amendemnt 4 = 1000

**Examples of how individual lines of the MT 707 should look:**

APPLICANT:  
OPERATOR:  
OPERATION DATE:  
OPERATION TIME:  
TEST KEY:  
BATCH TOTAL:  
SEGMENT TOTAL:  
MT/PRIORITY:707 02  
:27:1/1  
:20:10001981  
:21:1981  
:52D:Bank One  
100 Bank One Way  
Columbus ,OH 41984 US  
:31C:990204  
:30:990204  
:26E:1  
:59:David Fashion Creations P/L Pack  
Wholesale Division  
109 Ackland St.  
St. Kilda ,VA 30280-1234 US  
:32B:USD500,0  
:33B:USD0,0  
:34B:USD500,0  
:79:Letter of Credit: has been changed from 25 to 30  
for Style 10049369, resulting in an effect of 500  
(USD).

**The layout of the S.W.I.F.T MT 707 (Amendment to a Documentary Credit) file is as follows:**

SWIFT I.D. DATA TYPE CODES (refer to SWIFT User Handbook – Standards General Information – October 1998 release for formatting information):

**Notes:**

- The field lengths and types in the Oracle retail Standard Download Format of the MT 707 are important because sometimes they are different from the information that is being placed in them and the fields may have to be truncated, rounded, and so on.
- There is always a new line (nl) after every individual SWIFT ID (and there may be more than one line within an individual field (example 59 – Beneficiary, four lines to hold address information)).

- In some situations, certain fields are blank. These fields should be skipped over. In other words, no blank line or tag should be printed indicating the field is blank. Simply ignore it.

## Lcmt730 (SWIFT File Conversion - Letter of Credit Confirmation)

### Functional Area

Oracle Retail Trade Management – Letter of Credit Interfaces

### Module Affected

LCMT730

### Design Overview

The LCMT730 Perl script converts letter of credit confirmations from a S.W.I.F.T. format (MT730) to a RMS flat file format. The output file from this script is the input file for the LCUPLD.PC.

### Scheduling Constraints

| Schedule Information      | Description                                                              |
|---------------------------|--------------------------------------------------------------------------|
| Processing Cycle          | Phase 2                                                                  |
| Scheduling Considerations | LCMT730 should run prior to Letter of Credit upload program (lcupld.pc). |
| Pre-Processing            | N/A                                                                      |
| Post-Processing           | N/A                                                                      |
| Threading Scheme          | N/A                                                                      |

### Restart/Recovery

N/A

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

N/A

### I/O Specification

Input file format

| SWIFT I.D. and Description | Data type | Description                                              | How MT 730 fields are put into the RMS standard file format and what should be the size of RMS to be dealt with | Comments                                                                                                                                                                        |
|----------------------------|-----------|----------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 20 – Sender's Reference    | 16x       | LC number. The one assigned by the Sender (issuing bank) | FDETL<br>- Sender's reference, Char(16)                                                                         | This field maps to RTM's Bank LC Ref ID.                                                                                                                                        |
| 21 – Receiver's Reference  | 16x       | LC number assigned by the Receiver (retailer)            | FDETL<br>- Receiver's reference, Number(8) (NOREF used if unknown)                                              | This field maps to RTM's LC Ref ID. If this field has 'NOREF', the record must be rejected since this field is used to indicate the LC within RTM to which this record applies. |

|                                         |                                               |                                                                                                                                                                                                         |                                                     |                                                                                                                                                                                                                     |
|-----------------------------------------|-----------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 25 – Account Identification             | 35x                                           | Identifies the number of the account, which is used for the settlement of charges, on the books of the Sender.                                                                                          |                                                     | RTM currently does not have fields that map directly to this. Current position – is included in the input file. However, it is ignored during the upload process.                                                   |
| 30 – Date of Message Being Acknowledged | 6!n                                           | When a message is acknowledging a MT700, this field specifies the date of issue. In all other cases, this field specifies the date on which the message being acknowledged was sent.                    | FDETL<br>- Date of message Being Acknowledged, Date | This field maps to the LC activity date. As well, if this in confirming an LC application, it is mapped to the LC's confirmation date. Year interpretation: If YY>79 then YYMMDD = 19YYMMDD Else YYMMDD = 20YYMMDD. |
| 32a – Amount of Charges                 | Option B – 3!a15d<br><br>Option D – 6!n3!a15d | Contains the currency code and total amount of charges claimed by the sender of the message. When charges are debited, D is used (:32D) and when reimbursement for charges is needed, B is used (:32B). | FDETL<br>-Upload_type = 'C'onfirmation              | Current position – Because the 730 is only used for confirmations, this field does not contain any values. The upload type should be set equal to 'C'onfirmation.                                                   |

|                                     |                                                                                             |                                                                                                        |                                           |                                                                                                                                                                                                                                                           |
|-------------------------------------|---------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|-------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 57a – Account With Bank             | Option A –<br>[/1!a]/[34x]<br>4!a2!a2!c[<br>3!c]<br><br>Option D –<br>[/1!a]/[34x]<br>4*35x | This field specifies the bank to which the amount of charges is to be remitted in favor of the Sender. | FDETL<br>- Account With Bank,<br>Char(10) | Current position – is added to the input file however is ignored in the upload process. Because RTM has no facilities to maintain BICs or party identifiers, option D is always used for this field (that is, 57D) without [/1!a]/[34x] party identifier. |
| 71B – Charges                       | 6*35x                                                                                       | Specification of the charges claimed.                                                                  | FDETL<br>- Comments,<br>Char(2000)        | This field maps to RTM's activity comments field. Sender to Receiver information (72) is concatenated to this.                                                                                                                                            |
| 72 – Sender to Receiver Information | 6*35x                                                                                       | Text explanation if wanted.                                                                            | FDETL<br>- Comments,<br>Char(2000)        | This field maps to RTM's activity comments field. Charges (71B) is concatenated to this.                                                                                                                                                                  |

**Output File Layout**

| Record Name | Field Name                  | Field Type | Default Value                | Description                                                            |
|-------------|-----------------------------|------------|------------------------------|------------------------------------------------------------------------|
| File Header | File Type Record Descriptor | Char(5)    | FHEAD                        | Identifies file record type                                            |
|             | File Line Sequence Number   | Number(10) | specified by external system | Line number of the current file                                        |
|             | File Type Definition        | Char(4)    | LCUP                         | Identifies file as 'Letter of Credit Upload'                           |
|             | File Create Date            | Char (14)  | vdate                        | date file was written by external system<br>'YYYYMMDD HH24MISS' format |
| File Detail | File Type Record Descriptor | Char(5)    | FDETL                        | Identifies file record type                                            |



| Record Name  | Field Name                         | Field Type    | Default Value                | Description                                                                                                                                                                           |
|--------------|------------------------------------|---------------|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              | File Line Sequence Number          | Number(10)    | specified by external system | Line number of the current file                                                                                                                                                       |
|              | Sender's Reference                 | Char(16)      | lc_head.bank_ld_id           | The LC number that the bank assigns to a Letter of Credit.                                                                                                                            |
|              | Receiver's Reference               | Number(8)     | lc_activity.lc_ref_id        | The LC number that RMS assigned to the Letter of Credit.                                                                                                                              |
|              | Date of Message Being Acknowledged | Date (char 8) | lc_activity.activity_date    | If the upload type is 'L' then this date matches the date MT 700 date of issue (which we have not resolved between being the vdate or the lc_head.application_date) 'YYYYMMDD' format |
|              | Comments                           | Char(2000)    | lc_activity.comments         | Need to truncate? This field is probably a concatenation of the following SWIFT fields: 71B – Charges, 72 – Sender information.                                                       |
| File Trailer | File Type Record Descriptor        | Char(5)       | FTAIL                        | Identifies file record type                                                                                                                                                           |
|              | File Line Sequence                 | Number(10)    | Specified by external system | Line number of the current file                                                                                                                                                       |
|              | Total number of lines              | Number(10)    | Specified by external system | Total number lines in file                                                                                                                                                            |

## lcmt798 (SWIFT File Conversion – Letter of Credit Charges and Drawdowns)

### Functional Area

Oracle Retail Trade Management – Letter of Credit Interfaces

### Module Affected

LCMT798

## Design Overview

This Perl script converts letter of credit (L/C) activity data for charges and drawdowns from a S.W.I.F.T. format input file to a RMS format file.

## Scheduling Constraints

| Schedule Information      | Description                                                                                           |
|---------------------------|-------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 2                                                                                               |
| Scheduling Considerations | LCMT798 should be run prior to the Letter of Credit charges and drawings upload program (LCUP798.PC). |
| Pre-Processing            | N/A                                                                                                   |
| Post-Processing           | N/A                                                                                                   |
| Threading Scheme          | N/A                                                                                                   |

## Restart/Recovery

N/A

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| N/A   |        |        |        |        |

## I/O Specification

### Input File Layout

| Swift Tag                         | Description                                                                                                                    | Reqd? | Datatype                           | RMS Field                                         |
|-----------------------------------|--------------------------------------------------------------------------------------------------------------------------------|-------|------------------------------------|---------------------------------------------------|
| 20 – Transaction Reference Number | The sender's unambiguous identification of the transaction. Its detailed form and content are at the discretion of the sender. | Yes   | 16x – Transaction Reference Number | Bank L/C ID<br>Lc_head.bank_lc_id<br>Varchar2(16) |

| Swift Tag                         | Description                                                                                       | Reqd? | Datatype                                                                                                             | RMS Field                                                                                                                                                                                                                                                                                                                                                  |
|-----------------------------------|---------------------------------------------------------------------------------------------------|-------|----------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 12 – Type of Financial Instrument | This field classifies the financial instrument by a description or proprietary code.              | Yes   | Option A-<br>:4!c/[8c]/30x<br>:4!c – Qualifier<br>/ - Delimiter<br>[8c] – Issuer Code<br>/ - Delimiter<br>30x - Type | This field contains a constant identifier – ‘798’.                                                                                                                                                                                                                                                                                                         |
| 77E – Proprietary Message         | This field contains the proprietary message in a format agreed to by the Sender and the Receiver. | Yes   | Option E-<br>73x<br>[n*78x]                                                                                          | <p>This field contains the information below (fields 21, 23, 32C, 32D, 71A, 33A, 72).</p> <p>Carriage return, Line feed, Colon ‘CrLf:’ is used to separate fields included in this 77E.</p> <p><b>For example:</b></p> <p>:77E:’CrLf’</p> <p>:21:10004321:CrLf’</p> <p>:32C:990121USD1045</p> <p>etc...</p> <p>There may be multiple 77Es in one file.</p> |

| Swift Tag                   | Description                                                                                                                                                                                                                                                                                                    | Reqd? | Datatype | RMS Field                                                |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|----------|----------------------------------------------------------|
| 21 – Related Reference      | This field specifies, in an unambiguous way, a message or transaction identifier which is normally included as part of the information supplied with the message or transaction itself, and can subsequently be used to distinguish the message or transaction identified from other messages or transactions. | No    | 16x      | P/O Number<br><br>Lc_activity.order_no<br>Number(8)      |
| 23 – Further identification | This field specifies the type of transaction being confirmed, as well as the settlement method used.                                                                                                                                                                                                           | No    | 16x      | Invoice Number<br>Lc_activity.invoice_no<br>Varchar2(15) |

| Swift Tag             | Description                                                                                  | Reqd? | Datatype                                                               | RMS Field                                                                                                                                                                                                                                                                                                                                                              |
|-----------------------|----------------------------------------------------------------------------------------------|-------|------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 32C – Date and Amount | This field specifies the currency code and amount in a transaction and a corresponding date. | No    | Option C-6!n3!a15d<br><br>6!n – Date<br>3!a – Currency<br>15d – Amount | Charges Credited (this is interpreted as a positive amount).<br><br>Date is in format YYMMDD<br><br>The integer part of the Amount must contain at least one digit. A decimal comma ‘,’ is mandatory and is included in the maximum length.<br><br>Lc_activity.amount<br>Number(20,4)<br>Lc_activity.currency_code<br>Varchar2(3)<br>Lc_activity.activity_date<br>Date |
| 32D – Date and Amount | This field specifies the currency code and amount in a transaction and a corresponding date. | No    | Option D-6!n3!a15d<br><br>6!n – Date<br>3!a – Currency<br>15d – Amount | Charges Debited (this is interpreted as a negative amount)<br><br>Date is in format YYMMDD<br><br>The integer part of the Amount must contain at least one digit. A decimal comma ‘,’ is mandatory and is included in the maximum length.<br><br>Lc_activity.amount<br>Number(20,4)<br>Lc_activity.currency_code<br>Varchar2(3)<br>Lc_activity.activity_date<br>Date   |

| Swift Tag             | Description                                                                                  | Reqd? | Datatype                                                               | RMS Field                                                                                                                                                                                                                                                                                                                                                                              |
|-----------------------|----------------------------------------------------------------------------------------------|-------|------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 33A – Date and Amount | This field specifies the currency code and amount in a transaction and a corresponding date. | No    | Option A-6!n3!a15d<br><br>6!n – Date<br>3!a – Currency<br>15d – Amount | Date, currency, amount of drawing (this is interpreted as a positive amount)<br><br>Date is in format YYMMDD<br><br>The integer part of the Amount must contain at least one digit. A decimal comma ‘,’ is mandatory and is included in the maximum length.<br><br>Lc_activity.amount<br>Number(20,4)<br>Lc_activity.currency_code<br>Varchar2(3)<br>Lc_activity.activity_date<br>Date |
| 33C – Date and Amount | This field specifies the currency code and amount in a transaction and a corresponding date. | No    | Option A-6!n3!a15d<br><br>6!n – Date<br>3!a – Currency<br>15d – Amount | Date, currency, amount of drawing (this is interpreted as a negative amount)<br><br>Date is in format YYMMDD<br><br>The integer part of the Amount must contain at least one digit. A decimal comma ‘,’ is mandatory and is included in the maximum length.<br><br>Lc_activity.amount<br>Number(20,4)<br>Lc_activity.currency_code<br>Varchar2(3)<br>Lc_activity.activity_date<br>Date |

| Swift Tag                           | Description                                                                                                                                      | Reqd? | Datatype                                      | RMS Field                                         |
|-------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|-------|-----------------------------------------------|---------------------------------------------------|
| 72 – Sender to Receiver Information | This field specifies instructions or additional information for the Receiver, Intermediary, Account with Institution or Beneficiary Institution. | No    | 6*35x                                         | Comments<br>Lc_activity.comment<br>Varchar2(2000) |
| 18A – Number of Repetitive Parts    | This field specifies the number of times the repetitive part(s)/sequence (s) directly before or after this field appears in the message.         | No    | Option A-<br>5n – Number of Repetitive Parts. | Number of 77E's contained within the file.        |

## Output File Layout

| Record Name | Field Name                  | Field Type  | Default Value       | Description                                       |
|-------------|-----------------------------|-------------|---------------------|---------------------------------------------------|
| File Header | File Type Record Descriptor | Char(5)     | FHEAD               | Identifies file record type                       |
|             | File Line Identifier        | Number (10) | Line number in file | ID of current line being created for output file. |
|             | File Type Definition        | Char(4)     | LCCH                | Identifies file as 'Letter of Credit Changes'     |
|             | File Create Date            | Char(14)    | Create date         | Current date, formatted to 'YYYYMMDDHH24MISS'     |
| File Detail | File Type Record Descriptor | Char(5)     | FDETL               | Identifies file record type                       |
|             | File Line Sequence Number   | Number (10) | Line number in file | ID of current line being created for output file. |

| Record Name | Field Name                         | Field Type  | Default Value                                                                       | Description                                                                                                                                                                                                                                                                                  |
|-------------|------------------------------------|-------------|-------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Bank Letter of Credit Reference ID | Char(16)    | SWIFT tag 20                                                                        | Bank L/C ID                                                                                                                                                                                                                                                                                  |
|             | Order Number                       | Number (8)  | SWIFT tag 21                                                                        | Contains the order number that is attached to the letter of credit.                                                                                                                                                                                                                          |
|             | Invoice Number                     | Char (15)   | SWIFT tag 23                                                                        | Identifies the Issuing Bank's invoice number to which the drawdown refers. This field does not correspond to a RMS invoice number.                                                                                                                                                           |
|             | Transaction Number                 | Char (10)   | Null                                                                                | Identifies the amendment number or actual transaction number assigned by the bank.                                                                                                                                                                                                           |
|             | Transaction Code                   | Char (6)    | If the transaction is a Bank Charge – 'B'<br>If the transaction is a Drawdown – 'D' | Identifies the type of transaction that occurred.<br><br>The type is determined by what detail fields are received for the record. If the record contains a 33A this field stores get a 'D'. If the record contains either a 32C or 32D this field stores get a 'B'.                         |
|             | Amount Sign                        | Char (1)    | SWIFT 33A, 33C<br>SWIFT 32C, 32D                                                    | If the record contains a 33A field leave a blank space in this field.<br>If the record contains a 33C field this field should contain a '-'.<br>If the record contains a 32C field leave a blank space in this field.<br>If the record contains a 32D field this field should contain a '-'. |
|             | Amount                             | Number (20) | SWIFT 33A, 33C<br>SWIFT 32C, 32D                                                    | Holds the amount of the activity. This field has 4 implied decimal places.<br><br>If SWIFT 32C or 32D (Bank Charge) contains a value, use the amount from this field.<br><br>If SWIFT 33A or 33C (Drawdown) contains a value, use the amount from this field.                                |



| Record Name  | Field Name                  | Field Type  | Default Value                            | Description                                                                                                                                                                                                                          |
|--------------|-----------------------------|-------------|------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              | Currency Code               | Char (3)    | SWIFT 33A,<br>SWIFT 32C, 32D             | Contains the activity's currency code.<br>If SWIFT 32C or 32D (Bank Charge) contains a value, use the currency from this field.<br>If SWIFT 33A (Drawdown) contains a value, use the currency from this field.                       |
|              | Activity Date               | Char (8)    | SWIFT 33A,<br>SWIFT 32C, 32D             | Holds the date that the activity took place. Formatted to 'YYYYMMDD'<br>If SWIFT 32C or 32D (Bank Charge) contains a value, use the date from this field.<br>If SWIFT 33A (Drawdown) contains a value, use the date from this field. |
|              | Comments                    | Char (2000) | SWIFT tag 72                             | Holds any comments for the activity.                                                                                                                                                                                                 |
| File Trailer | File Type Record Descriptor | Char(5)     | FTAIL                                    | Identifies file record type                                                                                                                                                                                                          |
|              | File Line Identifier        | Number (10) | Sequential number<br>Created by program. | ID of current line being created for output file.                                                                                                                                                                                    |
|              | File Record Counter         | Number (10) |                                          | This contains the number of FDETL lines processed.                                                                                                                                                                                   |

## Icup798 (Letter of Credit Update)

### Functional Area

Oracle Retail Trade Management - Letter of Credit

### Module Affected

LCUP798.PC

### Design Overview

This program reads data from an input file containing letter of credit charges and drawings (in standard Oracle Retail format, modified from the SWIFT 798 format by the lcmt798 Perl script), validates it, and inserts it into the LC\_ACTIVITY table. If a record fails validation, it is written to a reject file. These rejected records can be reprocessed by Icup798 after errors are corrected.

## Scheduling Constraints

| Schedule Information      | Description                                 |
|---------------------------|---------------------------------------------|
| Processing Cycle          | Phase 2                                     |
| Scheduling Considerations | Should be run after the lcmt798 Perl script |
| Pre-Processing            | N/A                                         |
| Post-Processing           | N/A                                         |
| Threading Scheme          | N/A                                         |

## Restart/Recovery

This program is restartable but not threadable.

Restart/recovery logic for file-based processing is used. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| LC_HEAD        | Yes    | No     | No     | No     |
| LC_DETAIL      | Yes    | No     | No     | No     |
| LC_ACTIVITY    | No     | Yes    | No     | No     |
| LC_AMENDMENTS  | Yes    | No     | No     | No     |
| CURRENCIES     | Yes    | No     | No     | No     |
| CURRENCY_RATES | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |

## I/O Specification

The input file for this batch program is the output from the lcmt798 Perl script.

## Input File Layout

| Record Name | Field Name                         | Field Type  | Default Value     | Description                                                                                                              |
|-------------|------------------------------------|-------------|-------------------|--------------------------------------------------------------------------------------------------------------------------|
| FHEAD       | File head descriptor               | Char(5)     | FHEAD             | Describes file line type                                                                                                 |
|             | Line id                            | Number (10) | 0000000001        | Sequential file line number                                                                                              |
|             | File Type Definition               | Char(4)     | 'LCCH'            | Identifies as an LC 798 file-Letter of Credit Charges                                                                    |
|             | Current date                       | Date        |                   | File date in YYYYMMDDHH24MISS format                                                                                     |
| FDETL       | File record descriptor             | Char(5)     | FDETL             | Describes file line type                                                                                                 |
|             | Line id                            | Number (10) |                   | Sequential file line number                                                                                              |
|             | Bank letter of credit reference ID | Char (16)   | SWIFT tag 20      | Bank's LC ref ID                                                                                                         |
|             | Order number                       | Number(8)   | SWIFT tag 21      | Order number attached to LC.May be blank                                                                                 |
|             | Invoice number                     | Number (15) | SWIFT tag 23      | NOT a RMS invoice number, just a reference invoice number from the issuing bank. May be blank                            |
|             | Transaction number                 | Number (10) |                   | Amendment number or transaction number assigned by bank.May be null                                                      |
|             | Transaction code                   | Char(6)     | B or D            | 'B'ank charge or'D'rawdown                                                                                               |
|             | Amount                             | Number(21)  | SWIFT tag 33A,71A | (This is a 20-digit number with a leading – sign or blank and4 implied decimal places.)<br>Amount of charge or drawdown. |
|             | Currency code                      | Char(3)     | SWIFT 33A,71A     | Currency that the amount is in                                                                                           |
|             | Activity date                      | Date        | SWIFT 33A,32C,32D | Activity date(formatted as 'YYYYMMDD')                                                                                   |
|             | Comments                           | Char(2000)  | SWIFT tag 72      | Any comments associated with activity.May be null.                                                                       |

| Record Name | Field Name             | Field Type | Default Value | Description                                           |
|-------------|------------------------|------------|---------------|-------------------------------------------------------|
| FTAIL       | File record descriptor | Char(5)    | FTAIL         | Marks end of file                                     |
|             | Line id                | Char(10)   |               | Sequential file line number                           |
|             | Number of lines        | Number(10) |               | Number of lines in file not counting FHEAD and FTAIL. |

## Icupld (Letter of Credit Upload)

### Functional Area

Oracle Retail Trade Management

### Module Affected

LCUPLD.PC

### Design Overview

The LCUPLD program is used to upload LC (Letter of Credit) confirmations. After this program has processed a confirmation, the appropriate tables are updated; a confirmation updates the LC to confirm status and it writes the appropriate records to the LC\_ACTIVITY table.

### Scheduling Constraints

| Schedule Information      | Description          |
|---------------------------|----------------------|
| Processing Cycle          | Phase 2 (daily)      |
| Scheduling Considerations | N/A                  |
| Pre-Processing            | LCMT 730 Perl script |
| Post-Processing           | N/A                  |
| Threading Scheme          | N/A                  |

### Restart/Recovery

Restart/recovery for this program is set up at the individual FDETL record. Although there may be more than one FDETL record for a given LC, they stores each process as a separate entity.

File based restart/recovery must be used. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 10000 records.

### Locking Strategy

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table       | Select | Insert | Update | Delete |
|-------------|--------|--------|--------|--------|
| LC_HEAD     | Yes    | No     | Yes    | No     |
| LC_ACTIVITY | No     | Yes    | No     | No     |

**I/O Specification****Input File Layout**

| Record Name | Field Name                  | Field Type | Default Value         | Description                                                         |
|-------------|-----------------------------|------------|-----------------------|---------------------------------------------------------------------|
| File Header | File Type Record Descriptor | Char(5)    | FHEAD                 | Identifies file record type                                         |
|             | File Line Sequence Number   | Number(10) | 0000000001            | Line number of the current file                                     |
|             | File Type Definition        | Char(4)    | LCUP                  | Identifies file as 'Letter of Credit Upload'                        |
|             | File Create Date            | Char (14)  | vdate                 | Date file was written by external system<br>'YYMMDDHH24MISS' format |
| File Detail | File Type Record Descriptor | Char(5)    | FDETL                 | Identifies file record type                                         |
|             | File Line Sequence Number   | Number(10) |                       | Line number of the current file                                     |
|             | Sender's Reference          | Char(16)   | lc_head.bank_lc_id    | The LC number that the bank assigns to a Letter of Credit.          |
|             | Receiver's Reference        | Number(8)  | lc_activity.lc_ref_id | The LC number that Oracle Retail assigned to the Letter of Credit.  |

| Record Name  | Field Name                         | Field Type | Default Value             | Description                                                                                          |
|--------------|------------------------------------|------------|---------------------------|------------------------------------------------------------------------------------------------------|
|              | Date of Message Being Acknowledged | Char(14)   | lc_activity.activity_date | YYYYMMDDHH24MISS format                                                                              |
|              | Comments                           | Char(2000) | lc_activity.comments      | This field is a concatenation of the following SWIFT fields: 71B – Charges, 72 – Sender information. |
| File Trailer | File Type Record Descriptor        | Char(5)    | FTAIL                     | Identifies file record type                                                                          |
|              | File Line Sequence                 | Number(10) |                           | Line number of the current file                                                                      |
|              | Total number lines                 | Number(10) |                           | Total number of lines in file not including FHEAD and FTAIL                                          |

## tranupld (Transportation Upload)

### Functional Area

RTM

### Module Affected

TRANUPLD.PC

### Design Overview

This program uploads data from trading partners about the transportation of merchandise from the manufacturing site through customs clearance.

### Scheduling Constraints

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad Hoc      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

**Restart/Recovery**

The logical unit of work is a valid DTRAN record. The program reads each DTRAN record from the upload file, validates it and processes it. The recommended commit max counter value for this program is 1000 (this value depends on the implementation).

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| TRANSPORTATION | Yes    | Yes    | Yes    | Yes    |
| IF_ERRORS      | No     | Yes    | No     | No     |
| PARTNER        | Yes    | No     | No     | No     |
| FREIGHT_TYPE   | Yes    | No     | No     | No     |
| FREIGHT_SIZE   | Yes    | No     | No     | No     |
| CURRENCIES     | Yes    | No     | No     | No     |
| ORDHEAD        | Yes    | No     | No     | No     |
| ORDSKU         | Yes    | No     | No     | No     |
| ITEM_MASTER    | Yes    | No     | No     | No     |
| OUTLOC         | Yes    | No     | No     | No     |
| SCAC           | Yes    | No     | No     | No     |
| COUNTRY        | Yes    | No     | No     | No     |
| UOM_CLASS      | Yes    | No     | No     | No     |
| CODE_DETAIL    | Yes    | No     | No     | No     |

**I/O Specification**

Key : *Italicized field names* must be included in the input file.

**Input File Layout**

| Record Name | Field Name        | Field Type | Default Value | Description      |
|-------------|-------------------|------------|---------------|------------------|
| FTRAN       | Record descriptor | Char(5)    | FTRAN         | File head marker |

| Record Name | Field Name             | Field Type | Default Value | Description                                                                                                             |
|-------------|------------------------|------------|---------------|-------------------------------------------------------------------------------------------------------------------------|
|             | Line id                | Number(10) | 0000000001    | Unique line id                                                                                                          |
|             | File type definition   | Char(4)    | TRUP          | Identifies program as tranupld                                                                                          |
|             | File create date       | Char(14)   | Current date  | YYYYMMDDHHMISS format                                                                                                   |
| DTRAN       | Record descriptor      | Char(5)    | DTRAN         | Vessel, Voyage, ETD, Container, BL, Invoice File head                                                                   |
|             | Line id                | Number(10) |               | Unique line id                                                                                                          |
|             | Consolidator           | Char(10)   |               | Identifies the Consolidator. Validated against PARTNER table with type = 'CO'                                           |
|             | Vessel ID              | Char(20)   |               | Identifies the Vessel                                                                                                   |
|             | Voyage ID              | Char(10)   |               | Identifies the Voyage or Flight ID                                                                                      |
|             | Estimated Depart Date  | Char(8)    |               | YYYYMMDD format                                                                                                         |
|             | Shipment Number        | Char (20)  |               | Identifies an outside Shipment number                                                                                   |
|             | Actual Arrival Date    | Char(8)    |               | YYYYMMDD format                                                                                                         |
|             | Trans Mode             | Char(6)    |               | Identifies the type of transportation being used. Valid values are found in the TRMO Code Type on the CODE_DETAIL table |
|             | Vessel SCAC Code       | Char(6)    |               | Customs defined ID for the Vessel. Validated against SCAC table.                                                        |
|             | Estimated Arrival Date | Char(8)    |               | YYYYMMDD format                                                                                                         |
|             | Lading Port            | Char(5)    |               | Identifies the Lading Port. Validated against OUTLOC with type = 'LP'                                                   |
|             | Discharge Port         | Char(5)    |               | Identifies the Discharge Port. Validated against OUTLOC with type = 'DP'                                                |



| Record Name | Field Name              | Field Type     | Default Value           | Description                                                                                            |
|-------------|-------------------------|----------------|-------------------------|--------------------------------------------------------------------------------------------------------|
|             | Service Contract Number | Char(15)       |                         | Identifies the outside Service Contract Number                                                         |
|             | Container id            | Char(20)       |                         | Identifies the Container                                                                               |
|             | Container SCAC code     | Char(6)        |                         | Customs defined id for the container. Validated against SCAC table                                     |
|             | Delivery Date           | Char(8)        |                         | YYYYMMDD format                                                                                        |
|             | Seal id                 | Char(15)       |                         | Customs defined id for the container's seal                                                            |
|             | Freight Type            | Char(6)        |                         | Code that identifies the container type. Validated against the FREIGHT_TYPE table.                     |
|             | Freight Size            | Char(6)        |                         | Code that identifies the container size. Validated against the FREIGHT_SIZE table.                     |
|             | In Transit No.          | Char(15)       |                         | External transit number                                                                                |
|             | In Transit Date         | Char(8)        |                         | YYYYMMDD format                                                                                        |
|             | BL/AWB id               | Char(30)       |                         | Identifies the Bill of Lading or Air Way Bill                                                          |
|             | Candidate Ind           | Char(1)        | Defaulted to 'N'        | Identifies a complete Transportation record. Valid values are 'Y' and 'N'                              |
| DPOIT       | Record descriptor       | Char(5)        | DPOIT                   | Order/Item detail info                                                                                 |
|             | Line id                 | Number(10)     |                         | Unique file line id                                                                                    |
|             | ACD_Code                | Char(1)        |                         | Determines which process to perform 'A'dd, 'C'hange, 'D'elele.                                         |
|             | <i>Rush Ind</i>         | <i>Char(1)</i> | <i>Defaulted to 'N'</i> | <i>Identifies whether or not the item should be on a 'Rush' delivery. Valid values are 'Y' and 'N'</i> |
|             | Order number            | Number(8)      |                         | RMS order no                                                                                           |
|             | Item                    | Char(25)       |                         | RMS Item                                                                                               |

| Record Name | Field Name               | Field Type | Default Value | Description                                                                                   |
|-------------|--------------------------|------------|---------------|-----------------------------------------------------------------------------------------------|
|             | Invoice id               | Char(30)   |               | Identifies the Commercial Invoice                                                             |
|             | Invoice date             | Char(8)    |               | YYYYMMDD format                                                                               |
|             | Currency Code            | Char(3)    |               | Currency that the Currency Amount is reported in. Validated against CURRENCIES table.         |
|             | Exchange Rate            | Char (20)  |               | The exchange rate back to the primary currency (10 implied decimals)                          |
|             | Invoice amt              | Char 20)   |               | Invoice amt*10000 (with 4 implied decimal places), amount charged by supplier for the PO/Item |
|             | Origin Country id        | Char(3)    |               | Identifies where the PO/Item was made                                                         |
|             | Consolidation Country id | Char(3)    |               | Identifies where the PO/Items were consolidated                                               |
|             | Export Country id        | Char(3)    |               | Identifies where the PO/Items where shipped from                                              |
|             | Status                   | Char(6)    |               | Identifies the PO/Item status. Valid values are found in the TRCO Code Type on CODE_DETAIL    |
|             | Receipt ID               | Char(30)   |               | Identifies the external receipt number                                                        |
|             | FCR id                   | Char(15)   |               | Identifies the Freight Cargo Receipt id                                                       |
|             | FCR date                 | Char(8)    |               | YYYYMMDD format                                                                               |
|             | Packing Method           | Char(6)    |               | Identifies the Packing Type (Hanging or Flat). Valid values are 'HANG' or 'FLAT'              |
|             | Lot Number               | Char(15)   |               | Identifies the Lot Number of the PO/Item                                                      |
|             | Item Qty                 | Number(12) |               | Item Qty*10000(with 4 implied decimals), qty of Items                                         |
|             | Item QTY UOM             | Char(4)    |               | Identifies the UOM associated with the item quantity                                          |

| Record Name | Field Name     | Field Type | Default Value | Description                                                               |
|-------------|----------------|------------|---------------|---------------------------------------------------------------------------|
|             | Carton QTY     | Number(12) |               | Carton QTY*10000 (with 4 implied decimals), qty of Cartons                |
|             | Carton QTY UOM | Char(4)    |               | Identifies the UOM associated with the carton quantity                    |
|             | Gross WT       | Number(12) |               | Gross WT*10000 (with 4 implied decimals), Gross weight                    |
|             | Gross WT UOM   | Char(4)    |               | Identifies the UOM associated with the gross weight                       |
|             | Net WT         | Number(12) |               | Net WT*10000 (with 4 implied decimals), Net Weight                        |
|             | Net WT UOM     | Char(4)    |               | Identifies the UOM associated with the net weight                         |
|             | Cubic          | Number(12) |               | Cubic*10000 (with 4 implied decimals), cubic size                         |
|             | Cubic UOM      | Char(4)    |               | Identifies the UOM associated with the cubic size                         |
|             | Comments       | Char(256)  |               | User Comments                                                             |
| FTAIL       | Record type    | Char(5)    | FTAIL         |                                                                           |
|             | Line id        | Number(10) |               | Unique file line id                                                       |
|             | No. of lines   | Number(10) |               | Total number of transaction lines in file (not including FHEAD and FTAIL) |



---

## Organization Hierarchy Batch

### Overview

The organization hierarchy in RMS allows retailers to maintain records of all end locations (stores and warehouses), and to group those locations to meet the retailer's business and reporting needs. The structure of the organization hierarchy can be organized according to any arrangement a retailer requires, such as geography (west, central, south, and so on) or business type (mall, kiosk, and so on). Although the levels of the hierarchy are set in RMS, the content of each level is specific to the retailer.

RMS' organization hierarchy consists of:

- Company
- Chain
- Area
- Region
- District
- Store
- Warehouse

The levels of the organization hierarchy are numbered Level 0 through Level 5. Company (Level 0) is at the highest level of the organization hierarchy. The levels below company can be organized according to any arrangement your company requires. The lowest level, store (Level 5), defines where sales transactions occur. Although not tied to any particular level, warehouses are also included within the organization hierarchy.

### Organization Hierarchy Concepts

RMS uses the 'channel' concept in the organization hierarchy. In RMS' multi-channel option, locations have a stockholding property and a channel association. When adding a store, you associate the store with a channel. A store may be either a stockholding location, such as a virtual warehouse or a brick-and-mortar location, or a non-stockholding location, such as a Web store or catalog. This stockholding mechanism lets you set up inventory for non-stockholding sales channels in these virtual warehouses and then track sales by channel.

Actual physical warehouses are non-stockholding for RMS' purposes. RMS requires that all virtual warehouses must be associated with a physical warehouse. The physical warehouse serves as a grouping mechanism for one or more virtual warehouses. Each virtual warehouse is associated with a channel and considered a stockholding location. RMS can only 'see' these virtual warehouses, and external applications such as warehouse management systems are only aware of physical warehouses.

After multi-channel is set up, physical warehouse inventory, impacted by purchase orders and transfers for example, becomes apportioned to the virtual warehouses associated to that physical warehouse.

To determine if your implementation of RMS is set up to run multi-channel, look at the SYSTEM\_OPTIONS table's multichannel\_ind column for the value of 'Y' (yes). If the 'N' (no) value appears, multichannel is not enabled.

## Location Retail Overview

RMS requires all stores and warehouses to exist in location-level zone groups. This requirement means that RMS holds a zone level record against which a retail value can be stored.

## Batch Design Summary

The following batch designs are included in this functional area:

- LCLRBLD.PC (Location List Batch Rebuild)
- LIKESTORE.PC (Like Store Processing)
- SCHEDPRG.PC (Store Ship Schedule Purge)
- STOREADD.PC (Store Add)
- WHADD.PC (Warehouse Add)
- WHSTRASG (Warehouse Store Assignment)

## lclrbld (Location List Batch Rebuild)

### Functional Area

Location List

### Module Affected

LCLRBLD.PC

### Design Overview

The Location List Rebuild program performs the rebuilding of all location lists that have static\_ind = 'N' and batch\_rebuild\_ind = 'Y'. It calls package function LOCLIST\_BUILD\_SQL.REBUILD\_LIST to generate LOC\_LIST\_DETAIL records based on the store and warehouse criteria on the LOC\_LIST\_CRITERIA table.

### Scheduling Constraints

| Schedule Information      | Description                 |
|---------------------------|-----------------------------|
| Processing Cycle          | Ad-Hoc                      |
| Scheduling Considerations | N/A                         |
| Pre-Processing            | N/A                         |
| Post-Processing           | N/A                         |
| Threading Scheme          | Threaded by Location Lists. |

### Restart/Recovery

The logical unit of work for this program is a location list. The v\_restart\_loc\_list view is used for threading. Table-based restart/recovery is used by the batch program.

### Locking Strategy

The shared module LOCLIST\_BUILD\_SQL.REBUILD\_LIST locks the table LOC\_LIST\_HEAD for update and LOC\_LIST\_DETAIL for deletion.

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| LOC_LIST_HEAD   | Yes    | No     | Yes    | No     |
| LOC_LIST_DETAIL | Yes    | Yes    | No     | Yes    |

**I/O Specification**

N/A

**likestore (Store Add 'Like Store' Processing)****Functional Area**

Organization Hierarchy

**Module Affected**

LIKESTORE.PC

**Design Overview**

When a new store is created in RMS, there is an option to specify a “like” store. A like store serves as a template from which the new store being created is based on. The LIKESTORE batch is part of a sequence of batch programs that need to be run to ensure that a new store is successfully created in RMS when using the LIKESTORE option.

When the STOREADD batch is run, it sets the store open date and closes the date of all the like stores far in the future, so that those records are picked up in the LIKESTORE batch. The LIKESTORE batch then creates item-location relationships for all the items in the existing store with new store.

The LIKESTORE batch processes like stores and sets the store open and closes dates back to original date in the post process. The user can specify whether to copy the replenishment information, delivery schedules, and activity schedules from the existing store, which is copied in the LIKESTORE post process. It is necessary to run the STOREADD, LIKESTORE, and LIKESTORE post in the same order to successfully add all the stores in to RMS.

**Scheduling Constraints**

| Schedule Information      | Description              |
|---------------------------|--------------------------|
| Processing Cycle          | Ad-Hoc                   |
| Scheduling Considerations | N/A                      |
| Pre-Processing            | STOREADD.PC              |
| Post-Processing           | Prepost (likestore post) |

| Schedule Information | Description             |
|----------------------|-------------------------|
| Threading Scheme     | Threaded by department. |

### Restart/Recovery

The logical unit of work is store, item, pack indicator. The c\_add\_store cursor restarts the program based on store and the cursor c\_get\_items restart the program based on item, pack indicator. Threading is done by department using the v\_restart\_dept view.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| STORE_ADD              | Yes    | No     | No     | No     |
| STORE                  | Yes    | No     | No     | No     |
| PERIOD                 | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS         | Yes    | No     | No     | No     |
| COST_ZONE_GROUP        | Yes    | No     | No     | No     |
| ITEM_MASTER            | Yes    | No     | No     | No     |
| ITEM_EXP_HEAD          | Yes    | Yes    | No     | No     |
| PRICE_ZONE_GROUP_STORE | Yes    | No     | No     | No     |
| CLASS                  | Yes    | No     | No     | No     |
| ITEM_LOC               | Yes    | Yes    | No     | No     |
| ITEM_LOC_SOH           | No     | Yes    | No     | No     |
| PRICE_HIST             | No     | Yes    | No     | No     |
| ITEM_SUPP_COUNTRY_LOC  | Yes    | No     | No     | No     |
| ITEM_EXP_DETAIL        | Yes    | Yes    | No     | No     |

### I/O Specification

N/A



## schedprg (Store Ship Schedule Purge)

### Functional Area

Organizational Hierarchy

### Module Affected

SCHEDPRG.PC

### Design Overview

This program purges all old store ship scheule records and warehouse blackout records that exceeded the retention period as defined by the number of months value held in the ship\_sched\_history\_mths and loc\_close\_hist\_months columns on the SYSTEM\_OPTIONS table.

### Scheduling Constraints

| Schedule Information      | Description      |
|---------------------------|------------------|
| Processing Cycle          | Ad Hoc (Monthly) |
| Scheduling Considerations | N/A              |
| Pre-Processing            | N/A              |
| Post-Processing           | N/A              |
| Threading Scheme          | N/A              |

### Restart/Recovery

This program uses the commit\_max\_ctr on the RESTART\_CONTROL table to periodically commit delete operations. Periodic commits are performed to ensure that rollback segments are not exceeded in case of considerable volume.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS       | Yes    | No     | No     | No     |
| STORE_SHIP_DATE      | No     | No     | No     | Yes    |
| COMPANY_CLOSED_EXCEP | No     | No     | No     | Yes    |
| COMPANY_CLOSED       | No     | No     | No     | Yes    |

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| LOCATION_CLOSED | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**storeadd (Store Add)****Functional Area**

Location Retail

**Module Affected**

STOREADD.PC

**Design Overview**

This program adds all information necessary for a new store to function properly. When a store is added to the system, the store is accessible in the system only after storeadd.pc is run. The batch program loops through each record on the store\_add table and performs all the necessary inserts into the different RMS tables.

**Scheduling Constraints**

| Schedule Information      | Description                                    |
|---------------------------|------------------------------------------------|
| Processing Cycle          | AD HOC (daily)                                 |
| Scheduling Considerations | This batch should be run prior to likestore.pc |
| Pre-Processing            | N/A                                            |
| Post-Processing           | N/A                                            |
| Threading Scheme          | N/A                                            |

**Restart/Recovery**

After a record on store\_add is processed successfully, it is immediately deleted. Thus, restart recovery is implicit in storeadd.pc.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| PERIOD                  | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS          | Yes    | No     | No     | No     |
| STORE_ADD               | Yes    | No     | No     | Yes    |
| STORE                   | Yes    | Yes    | No     | No     |
| WH                      | Yes    | No     | No     | No     |
| POS_STORE               | Yes    | Yes    | No     | No     |
| PRICE_ZONE_GROUP        | Yes    | No     | No     | No     |
| PRICE_ZONE              | Yes    | Yes    | No     | No     |
| PRICE_ZONE_GROUP_STORE  | Yes    | Yes    | No     | No     |
| COST_ZONE_GROUP         | Yes    | No     | No     | No     |
| COST_ZONE               | Yes    | Yes    | No     | No     |
| COST_ZONE_GROUP_LOC     | Yes    | Yes    | No     | No     |
| SOURCE_DLVRY_SCHED      | Yes    | Yes    | No     | No     |
| SOURCE_DLVRY_SCHED_EXC  | Yes    | Yes    | No     | No     |
| SOURCE_DLVRY_SCHED_DAYS | Yes    | Yes    | No     | No     |
| COMPANY_CLOSED_EXCEP    | Yes    | Yes    | No     | No     |
| LOCATION_CLOSED         | Yes    | Yes    | No     | No     |
| STOCK_LEDGER_INSERTS    | No     | Yes    | No     | No     |
| LOC_TRAITS_MATRIX       | No     | Yes    | No     | No     |
| LOC_DISTRICT_TRAITS     | Yes    | No     | No     | No     |
| STORE_HIERARCHY         | No     | Yes    | No     | No     |
| REGIONALITY_TEMP        | No     | Yes    | No     | Yes    |
| REGIONALITY_MATRIX      | No     | Yes    | No     | Yes    |

**I/O Specification**

N/A

**whadd (Warehouse Add)****Functional Area**

Location Retail

**Module Affected**

WHADD.PC

## Design Overview

This batch program inserts pricing/zone information for new warehouses, virtual warehouses and/or internal finishers. It reads from the WH\_ADD table and inserts into PRICE\_ZONE and PRICE\_ZONE\_GROUP\_STORE for each retrieved record. Successfully processed records are deleted from the WH\_ADD table.

## Scheduling Constraints

| Schedule Information      | Description    |
|---------------------------|----------------|
| Processing Cycle          | AD HOC (daily) |
| Scheduling Considerations | N/A            |
| Pre-Processing            | N/A            |
| Post-Processing           | N/A            |
| Threading Scheme          | N/A            |

## Restart/Recovery

This program has inherent restart/recovery since records from WH\_ADD that were processed are deleted. As a result, the driving cursor never fetches the same record again. The logical unit of work is a location on WH\_ADD which can be a warehouse, virtual warehouses or internal finisher.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| PRICE_ZONE_GROUP       | Yes    | No     | No     | No     |
| WH_ADD                 | Yes    | No     | No     | Yes    |
| WH                     | Yes    | No     | No     | No     |
| PRICE_ZONE             | No     | Yes    | No     | No     |
| PRICE_ZONE_GROUP_STORE | No     | Yes    | No     | No     |

## I/O Specification

N/A

## whstrasg (Warehouse Store Assignment)

### Functional Area

Foundation Data

### Module Affected

WHSTRASG.PC

### Design Overview

This program is used to update the default warehouse attribute on the STORE table and the sourcing warehousing replenishment attribute on the REPL\_ITEM\_LOC table based on the warehouse-store assignments on the WH\_STORE\_ASSIGN table. The program is run every night and on the night before the assignment date for a warehouse-store assignment, it will update the relevant tables with the new warehouse-store relationship. If the warehouse is to be assigned to an item on the replenishment item location and the item is not associated to the warehouse, a non-fatal error message is raised indicating that the item is not associated to the warehouse. When updating the sourcing warehouse on the replenishment item location table, a system indicator (SYSTEM\_OPTIONS.wh\_store\_assign\_type) will determine whether the sourcing warehouse will be updated for warehouse-store replenishment records, cross-docked replenishment records, WH/cross link records, or all.

### Scheduling Constraints

| Schedule Information      | Description                                                              |
|---------------------------|--------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                          |
| Scheduling Considerations | Must be run after all replenishment batch programs.                      |
| Pre-Processing            | N/A                                                                      |
| Post-Processing           | prepost whstrasg post (clears the WH_STORE_ASSIGN of historical records) |
| Threading Scheme          | N/A                                                                      |

### Restart/Recovery

The logical unit of work is a store. A commit takes place when number of store records processed is equal to commit max counter from the RESTART CONTROL table. This program is restartable based on the last store successfully processed.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| WH_STORE_ASSIGN | Yes    | No     | Yes    | No     |
| STORE           | Yes    | No     | Yes    | No     |
| ITEM_LOC        | Yes    | No     | No     | No     |
| REPL_ITEM_LOC   | Yes    | No     | Yes    | No     |

### I/O Specification

N/A

---

## Point-of-Sale (POS) Download Batch

### Overview

RMS can update an external point-of-sale (POS) system for each POS store location.

The batch module used to accomplish the update is called POSDNLD.PC. POSDNLD.PC processes data contained in an RMS table called POS\_MODS that holds data that is written to it from three other RMS batch modules. The following is a description of the kinds of data that these modules insert into the POS\_MODS and how POSDNLD.PC transfers that data to the POS system. POSDNLD.PC is a template for custom interfaces.

The POS download batch module includes the following indicators in the file (from the POS\_MODS table)

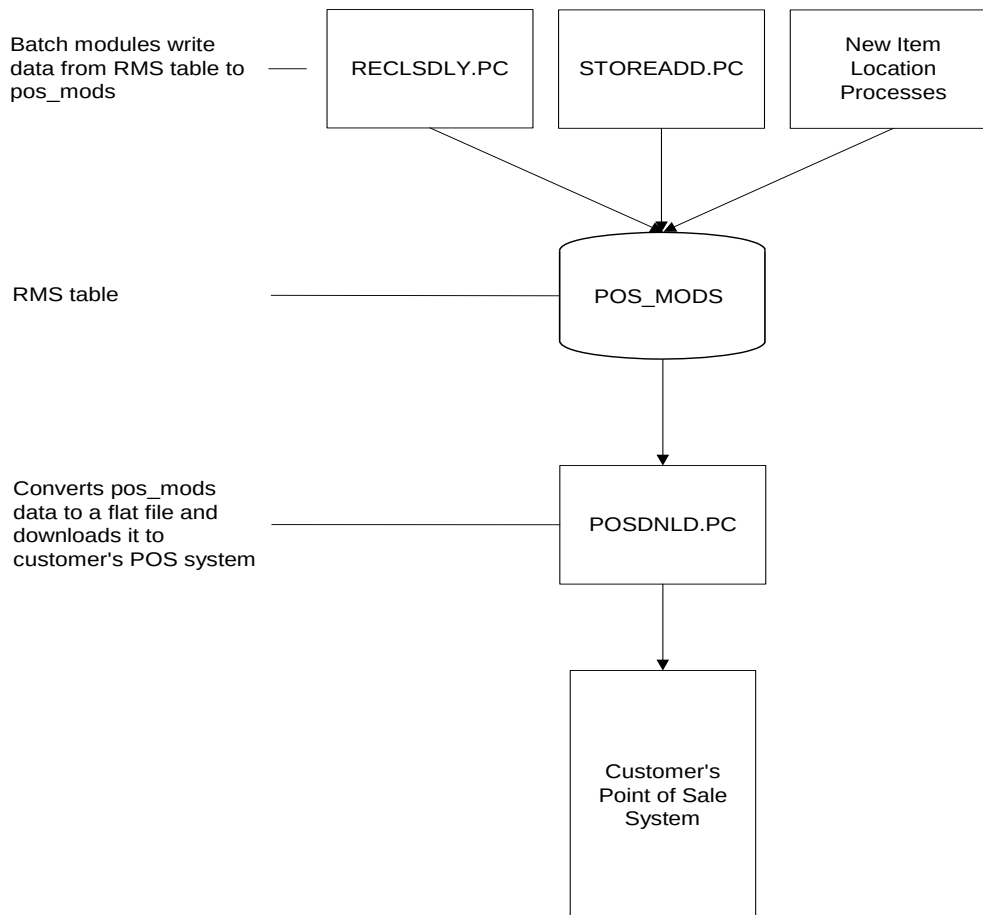
- CATCH\_WEIGHT\_IND (ITEM\_MASTER.CATCH\_WEIGHT\_IND)
- SALE\_TYPE (ITEM\_MASTER.SALE\_TYPE)

All deposit items are sent to the POS as per standard functionality, although if the item is a deposit item, it also has the linked deposit item number sent for contents items only.

## Point of Sale Download

The POS\_MODS table holds price updates that the batch module POSDNLD.PC outputs to a flat file for upload by the customer's point-of-sale (POS) application.

The following diagram illustrates the POS download process.



The POS Download Process

## Batch Design Summary

The following batch designs are included in this functional area:

- POSCDNLD.PC (Point of Sale Configuration Download)
- POSDNLD.PC (Point of Sale Download)
- POSGPDLD.PC (Group Number and Department Number Download to POS)

## poscdnld (Point Of Sale Configuration Download)

### Functional Area

Point of sale configuration

### Module Affected

POSCDNLD.PC



## Design Overview

This program handles the download of all 'POS configurations' from the RMS database to a flat file for loading into a POS/back office application. The program utilizes a series of table statuses to identify delta records to send down on a nightly basis, thus keeping interface volume down.

## Scheduling Constraints

| Schedule Information      | Description                                           |
|---------------------------|-------------------------------------------------------|
| Processing Cycle          | Phase 4                                               |
| Scheduling Considerations | This program should be run directly after POSDNLD.PC. |
| Pre-Processing            | N/A                                                   |
| Post-Processing           | poscdnld_post() – set status back to NULL             |
| Threading Scheme          | Single Thread                                         |

## Restart/Recovery

The logic unit of work is pos configuration type and pos configuration ID. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 1000 records (subject to change based on implementation).

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| POS_PAYINOUT_HEAD    | Yes    | No     | No     | Yes    |
| POS_TENDER_TYPE_HEAD | Yes    | No     | No     | Yes    |
| POS_BUTTON_HEAD      | Yes    | No     | No     | Yes    |
| POS_SUP_PAY_CRITERIA | Yes    | No     | No     | Yes    |
| POS_PROD_REST_HEAD   | Yes    | No     | No     | Yes    |
| POS_COUPON_HEAD      | Yes    | No     | No     | Yes    |
| POS_MONEY_ORD_HEAD   | Yes    | No     | No     | Yes    |
| POS_CONFIG_ITEMS     | No     | No     | No     | Yes    |

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| POS_STORE         | No     | No     | No     | Yes    |
| POS_DAY_TIME_DATE | No     | No     | No     | Yes    |
| SA_CC_VAL         | No     | No     | No     | Yes    |
| POS_ITEM_BUTTON   | No     | No     | No     | Yes    |
| POS_BUTTON_DETAIL | No     | No     | No     | Yes    |

## I/O Specification

This section includes all files layouts input and output.

### Output File Layout

| Record Name | Field Name          | Field Type    | Default Value                    | Description                                           |
|-------------|---------------------|---------------|----------------------------------|-------------------------------------------------------|
| FHEAD       | Record Type         | Char(5)       | 'FHEAD'                          | Record Identifier                                     |
|             | Line id             | Number(10)    | 0000000001                       | Sequential Line Identifier                            |
|             | File Name           | Char(4)       | 'POSC'                           | File Identifier                                       |
|             | File Date           | Char(14)      |                                  | Date the file was created in 'YYYYMMDD HHMMSS' format |
| TCOU        | Record Type         | Char(5)       | 'TCOUP'                          | Record Identifier                                     |
|             | Line id             | Number(10)    |                                  | Sequential Line Identifier                            |
|             | Coupon id           | Number(6)     |                                  |                                                       |
|             | Coupon Desc         | Char(250)     |                                  |                                                       |
|             | Currency Code       | Char2(3)      |                                  |                                                       |
|             | Max Discount Amount | Number(20, 4) |                                  |                                                       |
|             | Amount              | Number(20, 4) |                                  |                                                       |
|             | Percent Ind         | Char(1)       | 'N' – Amount<br>'Y' - Percentage |                                                       |
|             | Profit Center       | Char(6)       |                                  |                                                       |
|             | Tax Class           | Char(6)       |                                  |                                                       |

| Record Name  | Field Name      | Field Type | Default Value                                                                                                                                                                                    | Description                                                                                            |
|--------------|-----------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
|              | Export Code     | Char(6)    |                                                                                                                                                                                                  |                                                                                                        |
|              | Effective Date  | Char(14)   |                                                                                                                                                                                                  | Date the file was created in 'YYYYMMDD HHMMSS' format                                                  |
|              | Expiration Date | Char(14)   |                                                                                                                                                                                                  | Date the file would expire in the 'YYYYMMDD HHMMSS' format                                             |
|              | Prompted Ind    | Char(1)    | 'Y', 'N'                                                                                                                                                                                         | This indicator identifies if the cashier should be prompted to ask for a Coupon.                       |
|              | Display Ind     | Char(1)    | 'Y', 'N'                                                                                                                                                                                         | This indicator specifies whether the coupon is displayed in the list of valid coupons on the register. |
|              | Status          | Char(1)    | 'A','C','D'                                                                                                                                                                                      | Indicates if the POS configuration is new, is changed, or being deleted.                               |
|              | Vendor          | Number(10) |                                                                                                                                                                                                  |                                                                                                        |
|              | Vendor Type     | Char(6)    | 'AG' - Agent<br>'AP' - Applicant<br>'BK' - Bank<br>'BR' - Broker<br>'CN' - Coonsignee<br>'CO' - Consolidator<br>'FA' - Factory<br>'FF' - Freight Forwarder<br>'IM' - Importer<br>'SU' - Supplier |                                                                                                        |
|              | Promotion       | Number(10) |                                                                                                                                                                                                  |                                                                                                        |
|              | Coupon Barcode  | Char(20)   |                                                                                                                                                                                                  |                                                                                                        |
|              | Coupon Max Qty  | Number(6)  |                                                                                                                                                                                                  |                                                                                                        |
| <b>TPRES</b> | Record Type     | Char(5)    | 'TPRES'                                                                                                                                                                                          | Record Identifier                                                                                      |
|              | Line id         | Number(10) |                                                                                                                                                                                                  | Sequential Line Identifier                                                                             |

| Record Name | Field Name                   | Field Type    | Default Value                                                                                                                                                                                                                                                                    | Description                                           |
|-------------|------------------------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------|
|             | POS Product Restriction id   | Number(6)     |                                                                                                                                                                                                                                                                                  |                                                       |
|             | POS Product Restriction Desc | Char(120)     |                                                                                                                                                                                                                                                                                  |                                                       |
|             | POS Product Restriction Type | Char(6)       | 'PPRT' include:<br>'STMP' - Food Stamp<br>'MNAG' - Minimum Age<br>'CNDP' - Container Deposit<br>'CNVL' - Container Redemption Value<br>'DTDR' - Day/Time/Date Restriction<br>'TENT' - Tender Type<br>'NDSC' - Non-Discountable<br>'RTRN' - Returnable<br>'QLMT' - Quantity Limit |                                                       |
|             | Effective Date               | Char(14)      |                                                                                                                                                                                                                                                                                  | Date the file was created in 'YYYYMMDD HHMMSS' format |
|             | Currency Code                | Char(3)       |                                                                                                                                                                                                                                                                                  |                                                       |
|             | Product Restriction Amount   | Number(20, 4) |                                                                                                                                                                                                                                                                                  |                                                       |
|             | Age Minimum                  | Number(2)     |                                                                                                                                                                                                                                                                                  |                                                       |
|             | Date Restriction             | Char(14)      |                                                                                                                                                                                                                                                                                  | Date the file was created in 'YYYYMMDD HHMMSS' format |

| Record Name  | Field Name              | Field Type    | Default Value                                                                                                                                                                                                                                                                                                                                                    | Description                                                              |
|--------------|-------------------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------|
|              | Before Time Restriction | Char(6)       |                                                                                                                                                                                                                                                                                                                                                                  |                                                                          |
|              | After Time Restriction  | Char(6)       |                                                                                                                                                                                                                                                                                                                                                                  |                                                                          |
|              | Day Restriction         | Char(6)       |                                                                                                                                                                                                                                                                                                                                                                  |                                                                          |
|              | Max Qty Amount          | Number(12, 4) |                                                                                                                                                                                                                                                                                                                                                                  |                                                                          |
|              | Tender Type Group       | Char(6)       | 'CASH' - Cash,<br>'CHECK' - Check,<br>'CCARD' - Credit,<br>'COUPON' - Coupon,<br>'LOTTRY' - Lottery,<br>'FSTAMP' - Food Stamp,<br>'DCARD' - Debit Card,<br>'MORDER' - Money Order<br>'VOUCH' - Voucher<br>'ERR' - Error,<br>'SOCASS' - Social Assistance,<br>'TERM' - Termination Record,<br>'DRIVEO' - Drive Off,<br>'EBS' - Electronic Benefits ( Food Stamps) |                                                                          |
|              | Status                  | Char(1)       | 'A','C','D'                                                                                                                                                                                                                                                                                                                                                      | Indicates if the POS configuration is new, is changed, or being deleted. |
| <b>TMORD</b> | Record Type             | Char(5)       | 'TMORD'                                                                                                                                                                                                                                                                                                                                                          | Record Identifier                                                        |
|              | Line id                 | Number(10)    |                                                                                                                                                                                                                                                                                                                                                                  | Sequential Line Identifier                                               |

| Record Name   | Field Name       | Field Type    | Default Value | Description                                                                                                 |
|---------------|------------------|---------------|---------------|-------------------------------------------------------------------------------------------------------------|
|               | Money Order id   | Number(6)     |               |                                                                                                             |
|               | Money Order Desc | Char(120)     |               |                                                                                                             |
|               | Currency Code    | Char(3)       |               |                                                                                                             |
|               | Fee              | Number (20,4) |               |                                                                                                             |
|               | Max Face Value   | Number (20,4) |               |                                                                                                             |
|               | Max Sale Amount  | Number (20,4) |               |                                                                                                             |
|               | Tax Class        | Char (6)      |               |                                                                                                             |
|               | Cash Only Ind    | Char (1)      | ‘Y’, ‘N’      | The CASH_ONLY field identifies whether cash is the only method that can be used to purchase the Money Order |
|               | Serial Input Ind | Char (1)      | ‘Y’, ‘N’      | The SERIAL_INPUT_IND field specifies whether or not a serial input is associated with the Money Order.      |
|               | Pack Size        | Number (6)    |               |                                                                                                             |
|               | Refund Fee Ind   | Char (1)      | ‘Y’, ‘N’      | The REFUND_FEE_IND field specifies whether or not the Money Order Fee can be refunded.                      |
|               | Expiration Date  | Char(14)      |               | Date the file was created in ‘YYYYMMDD HHMMSS’ format                                                       |
|               | Effective Date   | Char(14)      |               | Date the file was created in ‘YYYYMMDD HHMMSS’ format                                                       |
|               | Status           | Char(1)       | ‘A’, ‘C’, ‘D’ | Indicates if the POS configuration is new, is changed, or being deleted.                                    |
| <b>TTTTYP</b> | Record Type      | Char(5)       | ‘TTTTYP’      | Record Identifier                                                                                           |

| Record Name | Field Name                 | Field Type    | Default Value                                                                                                                                                                                                             | Description                                           |
|-------------|----------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------|
|             | Line id                    | Number(10)    |                                                                                                                                                                                                                           | Sequential Line Identifier                            |
|             | Tender Type id             | Char(6)       |                                                                                                                                                                                                                           |                                                       |
|             | Tender Type Desc           | Char (120)    |                                                                                                                                                                                                                           |                                                       |
|             | Tender Type Group          | Char (6)      | 'CASH' - Cash,<br>'CHECK' - Check,<br>'CCARD' - Credit,<br>'COUPON' - Coupon,<br>'LOTTRY' - Lottery,<br>'FSTAMP' - Food Stamp,<br>'DCARD' - Debit Card,<br>'VOUCH' - Voucher,<br>'MORDER' - Money Order,<br>'ERR' - Error |                                                       |
|             | Effective Date             | Char(14)      |                                                                                                                                                                                                                           | Date the file was created in 'YYYYMMDD HHMMSS' format |
|             | Currency Code              | Char(3)       |                                                                                                                                                                                                                           |                                                       |
|             | Preset Amount              | Number(20, 4) |                                                                                                                                                                                                                           |                                                       |
|             | Authorize Minimum          | Number(20, 4) |                                                                                                                                                                                                                           |                                                       |
|             | Open Drawer Ind            | Char (1)      |                                                                                                                                                                                                                           |                                                       |
|             | Exact Change Ind           | Char (1)      |                                                                                                                                                                                                                           |                                                       |
|             | Accumulate Cash Intake Ind | Char (1)      |                                                                                                                                                                                                                           |                                                       |
|             | Next Dollar Ind            | Char (1)      |                                                                                                                                                                                                                           |                                                       |

| Record Name  | Field Name               | Field Type | Default Value | Description                                                              |
|--------------|--------------------------|------------|---------------|--------------------------------------------------------------------------|
|              | Deposit in Bank Ind      | Char (1)   |               |                                                                          |
|              | Deposit Override Ind     | Char (1)   |               |                                                                          |
|              | Automatic Deposit Ind    | Char (1)   |               |                                                                          |
|              | Pay In Deposit Ind       | Char (1)   |               |                                                                          |
|              | Ask For Invoice Ind      | Char (1)   |               |                                                                          |
|              | Imprint Ind              | Char (1)   |               |                                                                          |
|              | Imprint Tender Type      | Char (20)  |               |                                                                          |
|              | Show in Breakdown Ind    | Char (1)   |               |                                                                          |
|              | Display Ind              | Char (1)   |               |                                                                          |
|              | Discrepancy Display Type | Char (6)   |               |                                                                          |
|              | Processor Type           | Char (6)   |               |                                                                          |
|              | Export Code              | Char(6)    |               |                                                                          |
|              | Profit Center            | Char(6)    |               |                                                                          |
|              | Phone Authorization Type | Char(6)    |               |                                                                          |
|              | Status                   | Char(1)    | 'A','C','D'   | Indicates if the POS configuration is new, is changed, or being deleted. |
| <b>TBTTN</b> | Record Type              | Char(5)    | 'TBTTN'       | Record Identifier                                                        |
|              | Line id                  | Number(10) |               | Sequential Line Identifier                                               |
|              | Button configuration id  | Number (6) |               | Left-Justified Identifier                                                |



| Record Name  | Field Name         | Field Type | Default Value | Description                                                                                                                                                              |
|--------------|--------------------|------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              | Button Config Desc | Char(120)  |               |                                                                                                                                                                          |
|              | Button id          | Number (6) |               |                                                                                                                                                                          |
|              | Button Desc        | Char (120) |               |                                                                                                                                                                          |
|              | Layer Number       | Number (1) |               |                                                                                                                                                                          |
|              | Sequence Number    | Number (4) |               | Left-Justified Identifier                                                                                                                                                |
|              | Button Type        | Char(6)    |               | Valid Values include:<br>U = Multi Selection<br>S = Single Selection<br>I = Single Item<br>M = Major Category<br>N = Minor Category<br>X = No Button<br>P = Programmable |
|              | Effective Date     | Char(14)   |               |                                                                                                                                                                          |
|              | Text Color         | Char (6)   |               |                                                                                                                                                                          |
|              | Background Color   | Number (6) |               |                                                                                                                                                                          |
|              | Department         | Char(4)    |               |                                                                                                                                                                          |
|              | Text               | Char (10)  |               |                                                                                                                                                                          |
|              | Parent Button id   | Char(6)    |               |                                                                                                                                                                          |
|              | Status             | Char(1)    | 'A','C','D'   | Indicates if the POS configuration is new, is changed, or being deleted.                                                                                                 |
| <b>TPYIO</b> | Record Type        | Char(5)    | 'TPYIO'       | Record Identifier                                                                                                                                                        |
|              | Line id            | Number(10) |               | Sequential Line Identifier                                                                                                                                               |
|              | Pay In Out id      | Number(6)  |               |                                                                                                                                                                          |
|              | Pay In Out Desc    | Char(120)  |               |                                                                                                                                                                          |

| Record Name  | Field Name               | Field Type | Default Value                 | Description                                                                                    |
|--------------|--------------------------|------------|-------------------------------|------------------------------------------------------------------------------------------------|
|              | Pay In Out Type          | Char(1)    | 'I' – Pay In<br>'O' – Pay Out | Indicates whether the Pay In/Out type is a Pay In or a Pay Out.                                |
|              | Profit Center            | Char(6)    |                               |                                                                                                |
|              | Export Code              | Char(6)    |                               |                                                                                                |
|              | Tax Chart Code           | Char(6)    |                               |                                                                                                |
|              | Invoice Link             | Char(1)    | 'Y', 'N'                      | Indicates whether or not an invoice should be linked with this Pay In/Out type.                |
|              | Display Ind              | Char(1)    | 'Y', 'N'                      | Indicates whether the Pay In/Out configuration is displayed as a valid option in the register. |
|              | Effective Date           | Char(14)   |                               | Date the file was created in 'YYYYMMDD HHMMSS' format                                          |
|              | Status                   | Char(1)    | 'A','C','D'                   | Indicates if the POS configuration is new, is changed, or being deleted.                       |
| <b>TSPAY</b> | Record Type              | Char(5)    | 'TSPAY'                       | Record Identifier                                                                              |
|              | Line id                  | Number(10) |                               | Sequential Line Identifier                                                                     |
|              | Supplier                 | Number(10) |                               |                                                                                                |
|              | Pay Type                 | Char(6)    |                               | Valid Values include:<br>CASH = Cash<br>MP = Money Order<br>INVC = Invoice                     |
|              | Supplier Payment Type id | Number(6)  |                               |                                                                                                |
|              | Effective Date           | Char(14)   |                               | Date the file was created in 'YYYYMMDD HHMMSS' format                                          |
|              | Status                   | Char(1)    | 'A','C','D'                   | Indicates if the POS configuration is new, is changed, or being deleted.                       |

| Record Name  | Field Name  | Field Type | Default Value                             | Description                                                                                                               |
|--------------|-------------|------------|-------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| <b>TSTOR</b> | Record Type | Char(5)    | 'TSTOR'                                   | Record Identifier                                                                                                         |
|              | Line id     | Number(10) |                                           | Sequential Line Identifier                                                                                                |
|              | Store       | Number(10) |                                           | Left-Justified Store Identifier                                                                                           |
|              | Status      | Char(1)    | 'A' – Add<br>'D' – Delete<br>'C' - Change | Indicates to the POS if the POS configuration must be added or deleted or changed.                                        |
| <b>TITEM</b> | Record Type | Char(5)    | 'TITEM'                                   | Record Identifier                                                                                                         |
|              | Line id     | Number(10) |                                           | Sequential Line Identifier                                                                                                |
|              | Item        | Char(25)   |                                           | Left-Justified Item Identifier                                                                                            |
|              | Status      | Char(1)    | 'A' – Add<br>'D' – Delete<br>'C' – Change | Indicates the item's status at the POS. Overlays of items as a result of a change to the merch criteria has a 'C' status. |
|              | Button id   | Number(6)  | id                                        | Indicates the identification number of the button.                                                                        |

## posdnld (Point Of Sale Download)

### Functional Area

Point of Sale Download

### Module Affected

POSDNLD.PC

### Design Overview

The POSDNLD program is used to download pos\_mods records created in the RMS to the store Point-of-Sale (POS) systems. All item-level or item/location-level changes are sent to the POS via POSDNLD, including new item locations, department-class-subclass changes, and item location traits. The output file of this program contains all records for all stores in a given run.

### Scheduling Constraints

| Schedule Information | Description |
|----------------------|-------------|
| Processing Cycle     | Ad hoc      |

| Schedule Information      | Description                                                                                                         |
|---------------------------|---------------------------------------------------------------------------------------------------------------------|
| Scheduling Considerations | This program is run towards the end of the batch run when all pos_mods records are created for the transaction day. |
| Pre-Processing            | N/A                                                                                                                 |
| Post-Processing           | prepost.pc - posdnld_post() – records in POS_MODS are truncated.                                                    |
| Threading Scheme          | Threaded by store                                                                                                   |

### Restart/Recovery

The logical unit of work for this program is set at the store/item level. Threading is done by store using the v\_restart\_store view to thread properly.

Both table and file restart/recovery must be used. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 10000 records (subject to change based on implementation).

### Locking Strategy

N/A

### Security Considerations

Price changes for all stores are stored in a Unix file with the processes default permissions (umask). Care should be exercised so that this file cannot be tampered with.

### Performance Considerations

N/A

### Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| POS_MODS        | Yes    | No     | No     | No     |
| ITEM_MASTER     | Yes    | No     | No     | No     |
| ITEM_LOC        | Yes    | No     | No     | No     |
| V_RESTART_STORE | Yes    | No     | No     | No     |

## I/O Specification

### Output flat file specification

When the data is sent to the POS via POSDNLD.PC, the regular/clearance price indicator is included in the download file shown below.

All input comes from the POS\_MODS table. All columns of this table can be NULL with the exception of tran\_type and store. Most columns should default to blank (spaces) with the exception of:

- new\_price, new\_multi\_units, new\_multi\_units\_retail, proportional\_tare\_pct and fixed\_tare\_value. These should default to zero (0).
- start\_date, start\_time and end\_time. These should default to period.vdate + 1.

| Record Name | Field Name                  | Field Type    | Default Value                             | Description                                                                                                                                                                                                                                                                         |
|-------------|-----------------------------|---------------|-------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| File Header | File Type Record Descriptor | Char(5)       | FHEAD                                     | Identifies file record type.                                                                                                                                                                                                                                                        |
|             | File Line Identifier        | Number ID(10) | 0000000001                                | ID of current line being created for output file.                                                                                                                                                                                                                                   |
|             | File Type Definition        | Char(4)       | POSD                                      | Identifies file as 'POS Download'.                                                                                                                                                                                                                                                  |
|             | File Create Date            | Char(8)       |                                           | Vdate, formatted to 'YYYYMMDD'.                                                                                                                                                                                                                                                     |
| File Detail | File Type Record Descriptor | Char(5)       | FDETL                                     | Identifies file record type.                                                                                                                                                                                                                                                        |
|             | File Line Identifier        | Number ID(10) | Sequential number.<br>Created by program. | ID of current line being created for output file.                                                                                                                                                                                                                                   |
|             | Location Number             | Number (10)   | Store                                     | Contains the store location that is affected by the transaction.                                                                                                                                                                                                                    |
|             | Update Type                 | Char(1)       | Update type.<br>Created by program.       | Code used for retailer specific POS system.<br>1 - Transaction Types 1 & 2.<br>2 - Transaction Types 10 thru 18, 31 and 32, 50 thru 57, 59 thru 64.<br>3 - Transaction Types 21 & 22<br>4 - Transaction Types 25 & 26<br>0 - All other Transaction Types. These should never exist. |

| Record Name | Field Name       | Field Type | Default Value                       | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-------------|------------------|------------|-------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Start Date       | Char(8)    | Start_date or vdate + 1 if NULL.    | The effective date for the action determined by the transaction type of the record. Formatted to 'YYYYMMDD'.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|             | Time             | Char(6)    | Start_time, End_time or start_date. | This field is used in conjunction with starting a promotion (Transaction Type = 31). Start time indicates the time of day that the promotion is scheduled to start. This field is also used in conjunction with ending a promotion (Transaction Type = 32). Any other Transaction Type uses the time from the start_date column. Formatted to 'HH24MISS'.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|             | Transaction Type | Number(2)  | Tran_type                           | Indicates the type of transaction to determine what Oracle Retail action is being sent down to the stores from the Oracle Retail POS_MODS table.<br>Valid values include:<br>01 - Add new transaction level item<br>02 - Add new lower than transaction level item<br>10 - Change Short Description of existing item<br>12 - Change Description of an existing item<br>13 - Change Department/Class/Subclass of an existing item<br>20 - Change in VAT rate<br>21 - Delete existing transaction level item<br>22 - Delete existing lower than transaction level item<br>25 - Change item's status<br>26 - Change item's taxable indicator<br>50 - Change item's launch date<br>51 - Change item's quantity key options<br>52 - Change item's manual price entry options<br>53 - Change item's deposit code<br>54 - Change item's food stamp |

| Record Name | Field Name                 | Field Type | Default Value        | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------|----------------------------|------------|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             |                            |            |                      | <p>indicator</p> <p>55 - Change item's WIC indicator</p> <p>56 - Change item's proportional tare percent</p> <p>57 - Change item's fixed tare value</p> <p>58 - Change item's rewards eligible indicator</p> <p>59- Change item's electronic marketing clubs</p> <p>60 - Change item's return policy</p> <p>61 - Change item's stop sale indicator</p> <p>62 – Change item's returnable indicator</p> <p>63 – Change item's refundable indicator</p> <p>64 – Change item's back order indicator</p> <p>65 – Change items deposit linked item</p> |
|             | Item Number ID             | Char(25)   | Item                 | This field identifies the unique alphanumeric value for the transaction level item. The ID number of an item from the Oracle Retail ITEM_MASTER table.                                                                                                                                                                                                                                                                                                                                                                                           |
|             | Item Number Type           | Char(6)    | Item_number_type     | This field identifies the type of the item number ID.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|             | Format ID                  | Char(1)    | Format_id            | This field identifies the type of format used if the item_number_type is 'VPLU'.                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|             | Prefix                     | Number(2)  | Prefix               | This field identifies the prefix used if the item_number_type is 'VPLU'. In case of single digit prefix, the field is right-justified with blank padding.                                                                                                                                                                                                                                                                                                                                                                                        |
|             | Reference Item             | Char(25)   | Ref_item             | This field identifies the unique alphanumeric value for an item one level below the transaction level item.                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|             | Reference Item Number Type | Char(6)    | Ref_item_number_type | This field identifies the type of the ref item number ID.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

| Record Name | Field Name               | Field Type    | Default Value          | Description                                                                                                                                                                                                                                         |
|-------------|--------------------------|---------------|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Reference Item Format ID | Char(1)       | Ref_format_id          | This field identifies the type of format used if the ref item_number_type is 'VPLU'.                                                                                                                                                                |
|             | Reference Item Prefix    | Number (2)    | Ref_prefix             | This field identifies the prefix used if the ref item_number_type is 'VPLU'. In case of single digit prefix, the field is right-justified with blank padding.                                                                                       |
|             | Item Short Description   | Char(120)     | Item_short_desc        | Contains the short description associated with the item.                                                                                                                                                                                            |
|             | Item Long Description    | Char (250)    | Item_long_desc         | Contains the long description associated with the item.                                                                                                                                                                                             |
|             | Department ID            | Number (4)    | Dept                   | Contains the item's associated department.                                                                                                                                                                                                          |
|             | Class ID                 | Number (4)    | Class                  | Contains the item's associated class.                                                                                                                                                                                                               |
|             | Subclass ID              | Number (4)    | Subclass               | Contains the item's associated subclass.                                                                                                                                                                                                            |
|             | New Price                | Number (20,4) | New_price              | Contains the new effective price in the selling unit of measure for an item when the transaction type identifies a change in price. Otherwise, the current retail price is used to populate this field. This field is stored in the local currency. |
|             | New Selling UOM          | Char(4)       | New_selling_uom        | Contains the new selling unit of measure for an item's single-unit retail.                                                                                                                                                                          |
|             | New Multi Units          | Number (12,4) | New_multi_units        | Contains the new number of units sold together for multi-unit pricing. This field is only filled when a multi-unit price change is being made.                                                                                                      |
|             | New Multi Units Retail   | Number (20,4) | New_multi_units_retail | Contains the new price in the selling unit of measure for units sold together for multi-unit pricing. This field is only filled when a multi-unit price change is being made. This field is stored in the local currency.                           |
|             | New Multi Selling UOM    | Char(4)       | New_multi_selling_uom  | Contains the new selling unit of measure for an item's multi-unit retail.                                                                                                                                                                           |



| Record Name | Field Name        | Field Type  | Default Value | Description                                                                                                                                                                                                                                                                                |
|-------------|-------------------|-------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Status            | Char(1)     | Status        | <p>Populates if tran_type for the item is 1(new item added) or 25 (change item status) or 26 (change taxable indicator).</p> <p>Contains the current status of the item at the store.</p> <p>Valid values are:<br/> A = Active<br/> I = Inactive<br/> D = Delete<br/> C = Discontinued</p> |
|             | Taxable Indicator | Char(1)     | Taxable_ind   | <p>Populates if tran_type for the item is 1 (new item added) or 25 (change item status) or 26 (change taxable indicator).</p> <p>Indicates whether the item is taxable at the store. Valid values are 'Y' or 'N'.</p>                                                                      |
|             | Promotion Number  | Number (10) | Promotion     | <p>This field contains the number of the promotion for which the discount originated. This field, along with the Mix Match Number or Threshold Number is used to isolate a list of items that tie together with discount information.</p>                                                  |
|             | Mix Match Number  | Number (10) | Mix_match_no  | <p>This field contains the number of the mix and match in a promotion for which the discount originated. This field, along with the promotion, is used to isolate a list of items which tie together with the mix and match discount information.</p>                                      |

| Record Name | Field Name           | Field Type  | Default Value      | Description                                                                                                                                                                                                                                                                                        |
|-------------|----------------------|-------------|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Mix Match Type       | Char(1)     | Mix_match_type     | This field identifies which types of mix and match record this item belongs to. The item can either be a buy (exists on PROM_MIX_MATCH_BUY) or a get (exists on PROM_MIX_MATCH_GET) item. This field is only populated when the MIX_MATCH_NO is populated. Valid values are:<br>B - Buy<br>G – Get |
|             | Threshold Number     | Number (10) | Threshold_no       | This field contains the number of the threshold in a promotion for which the discount originated. This field, along with the promotion, is used to isolate a list of items that tie together with discount information.                                                                            |
|             | Launch Date          | Char(8)     | Launch_date        | Date that the item should first be sold at this location, formatted to 'YYYYMMDD'.                                                                                                                                                                                                                 |
|             | Quantity Key Options | Char(6)     | Qty_key_options    | Determines whether the price can/should be entered manually on a POS for this item at the location. Valid values are in the code_type 'RPO'. Current values include 'R - required', 'P - Prohibited'.                                                                                              |
|             | Manual Price Entry   | Char(6)     | Manual_price_entry | Determines whether the price can/should be entered manually on a POS for this item at the location. Valid values are in the code_type 'RPO'. Current values include 'R - required', 'P - Prohibited', and 'O - Optional'.                                                                          |

| Record Name | Field Name                | Field Type    | Default Value         | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------|---------------------------|---------------|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Deposit Code              | Char(6)       | Deposit_code          | Indicates whether a deposit is associated with this item at the location. Valid values are in the code_type 'DEPO'. Additional values may be added or removed as needed. Deposits are not subtracted from the retail of an item uploaded to RMS, etc. This kind of processing is the responsibility of the retailer and should occur before sales are sent to any Oracle Retail application.                                                                                                                                                                    |
|             | Food Stamp Indicator      | Char(1)       | Food_stamp_ind        | Indicates whether the item is approved for food stamps at the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|             | WIC Indicator             | Char(1)       | Wic_ind               | Indicates whether the item is approved for WIC at the location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|             | Proportional Tare Percent | Number (12,4) | Proportional_tare_pct | Holds the value associated of the packaging in items sold by weight at the location. The proportional tare is the proportion of the total weight of a unit of an item that is packaging (for example, if the tare item is bulk candy, this is the proportional of the total weight of one piece of candy that is the candy wrapper). The only processing RMS does involving the proportional tare percent is downloading it to the POS.                                                                                                                         |
|             | Fixed Tare Value          | Number (12,4) | Fixed_tare_value      | Holds the value associated of the packaging in items sold by weight at the location. Fixed tare is the tare of the packaging used to (for example, if the tare item is bulk candy, this is weight of the bag and twist tie). The only processing RMS does involving the fixed tare value is downloading it to the POS. Fixed tare is not subtracted from items sold by weight when sales are uploaded to RMS, and so on. This kind of processing is the responsibility of the retailer and should occur before sales are sent to any Oracle Retail application. |

| Record Name | Field Name                | Field Type     | Default Value       | Description                                                                                                                                                                                                                                                                                                                       |
|-------------|---------------------------|----------------|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Fixed Tare UOM            | Char(4)        | Fixed_tare_uom      | Holds the unit of measure value associated with the tare value. The only processing RMS does involving the proportional tare value and unit of measure (UOM) is downloading it to the POS. This kind of processing is the responsibility of the retailer and should occur before sales are sent to any Oracle Retail application. |
|             | Reward Eligible Indicator | Char(1)        | Reward_eligible_ind | Holds whether the item is legally valid for various types of bonus point/award programs at the location.                                                                                                                                                                                                                          |
|             | Elective Marketing Clubs  | Char(6)        | Elect_mtk_clubs     | Holds the code that represents the marketing clubs to which the item belongs at the location. Valid values can belong to the code_type 'MTKC'. Additional values can be added or removed from the code type as needed                                                                                                             |
|             | Return Policy             | Char(6)        | Return_pocily       | Holds the return policy for the item at the location. Valid values for this field belong to the code_type 'RETP'.                                                                                                                                                                                                                 |
|             | Stop Sale Indicator       | Char(1)        | Stop_sale_ind       | Indicates that sale of the item should be stopped immediately at the location (for example, in case of recall, and so on).                                                                                                                                                                                                        |
|             | Returnable Indicator      | Char(1)        | Returnable_ind      | Indicates that the item is returnable at the location when equal to 'Y'es. Indicates that the item is not returnable at the location when equal to 'N'o.                                                                                                                                                                          |
|             | Refundable Indicator      | Char(1)        | Refundable_ind      | Indicates that the item is refundable at the location when equal to 'Y'es. Indicates that the item is not refundable at the location when equal to 'N'o.                                                                                                                                                                          |
|             | Back Order Indicator      | Char(1)        | Back_order_ind      | Indicates that the item is back orderable at the location when equal to 'Y'. Indicates that the item is not back orderable when equal to 'N'o.                                                                                                                                                                                    |
|             | Vat Code                  | Char(6)        |                     | Indicates the VAT code used with this item.                                                                                                                                                                                                                                                                                       |
|             | Vat Rate                  | Number (20,10) |                     | Indicates the VAT rate associated with this item and VAT code.                                                                                                                                                                                                                                                                    |

| Record Name  | Field Name                  | Field Type | Default Value  | Description                                                                                                               |
|--------------|-----------------------------|------------|----------------|---------------------------------------------------------------------------------------------------------------------------|
|              | Class Vat Indicator         | Char(1)    |                | Indicates whether or not the class VAT indicator is on or off for the class that this item exists in.                     |
|              | Promotion Item Type         | Char(1)    | Prom_item_type | Indicates the type of items where the promotion should apply. Valid values for this field belong to the code_type 'PREM.' |
|              | CATCH_WEIGHT_IN D           | Char(1)    |                | Indicator whether or not an item is a catch weight item                                                                   |
|              | SALE_TYPE                   | CHAR(6)    |                | Set-up of the item at the time of sale. Valid values are:<br>V – variable weight each<br>L – loose weight                 |
|              | CONTAINER_ITEM              | CHAR(25)   |                | Linked container item number for a contents item                                                                          |
| File Trailer | File Type Record Descriptor | Char(5)    | FTAIL          | Identifies file record type                                                                                               |
|              | Line number                 | Number(10) |                | Sequential file line number (total # lines in file)                                                                       |
|              | Number of transactions      | Number(10) |                | Number of transactions in file                                                                                            |

## posgpdl (Group Number and Department Number Download to POS)

### Functional Area

POS Configuration

### Module Affected

POSGPDL.PC

### Design Overview

The POSGPDL.PC program is used to download the group number and department information to the Point of Sale (POS) system. The retailer uses this information when sale of merchandise is done at department level.

## Scheduling Constraints

| Schedule Information      | Description                                                                                                                                    |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 4 (Daily)                                                                                                                                |
| Scheduling Considerations | Run at the end of a batch cycle, after the program RECLSDLY.PC (reclassify items from one department, class or subclass to other) is complete. |
| Pre-Processing            | N/A                                                                                                                                            |
| Post-Processing           | N/A                                                                                                                                            |
| Threading Scheme          | N/A                                                                                                                                            |

## Restart/Recovery

The logical unit of work for this program is set at the group number and department level. Threading is not done on this program.

The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O. The recommended commit counter setting is 10 records (subject to change based on implementation).

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

The assumption is that the dataset that defines group numbers and departments is small. With this assumption, a full extract of RMS group number and department number occurs during every run of the batch program.

## Key Tables Affected

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| DEPS  | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name           | Field Type | Default Value | Description                 |
|-------------|----------------------|------------|---------------|-----------------------------|
| FHEAD       | File head descriptor | Char(5)    | FHEAD         | Describes file line type    |
|             | Line id              | Char(10)   | 0000000001    | Sequential file line number |
|             | File Type ID         | Char(4)    | 'PGPD'        | File identifier             |

| Record Name | Field Name             | Field Type  | Default Value | Description                                          |
|-------------|------------------------|-------------|---------------|------------------------------------------------------|
|             | Current date           | Char(14)    |               | File date in YYYYMMDDHH24MISS format                 |
| FDETL       | File record descriptor | Char(5)     | FDETL         | Describes file line type                             |
|             | Line id                | Number (10) |               | Sequential file line number.                         |
|             | Group Number           | Number (4)  |               | RMS Group Number.                                    |
|             | Department Number      | Number (4)  |               | RMS Department number.                               |
| FTAIL       | File record descriptor | Char(5)     | FTAIL         | Marks end of file                                    |
|             | Line id                | Char(10)    |               | Sequential file line number                          |
|             | Number of lines        | Number(10)  |               | Number of lines in file not counting FHEAD and FTAIL |

## Overview

The Prepost batch program helps facilitate the multi-threading capabilities of other batch programs within RMS.

Due to the nature of the threading algorithm, individual programs might need a pre or a post program run to initialize variables or files before any of the threads have run or to update final data once all the threads are run. The decision was made to create pre-programs and post-programs in these cases rather than let the restart/recovery logic decide whether the currently processed thread is the first thread to start or the last thread to end for a given program.

## Batch Design Summary

The following batch design is included in this functional area:

- PREPOST.PC (Pre/Post)

## prepost (Pre/Post)

### Functional Area

Pre/Post Functionality

### Module Affected

PREPOST.PC

### Design Overview

The pre/post module facilitates multi-threading by allowing general system administration functions (such as table deletions or mass updates) to be completed after all threads of a particular program are processed.

This program takes three parameters: username/password to log on to Oracle, a program before or after which this script must run and an indicator telling whether the script is a pre or post function. It acts as a shell script for running all pre-program and post-program updates and purges (the logic was removed from the programs themselves to enable multi-threading and restart/recovery).

### Scheduling Constraints

| Schedule Information      | Description        |
|---------------------------|--------------------|
| Processing Cycle          | All phases (daily) |
| Scheduling Considerations | N/A                |
| Pre-Processing            | N/A                |
| Post-Processing           | N/A                |
| Threading Scheme          | N/A                |



**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                                | Select | Insert | Update | Index | Delete | Truncate | Trigger |
|--------------------------------------|--------|--------|--------|-------|--------|----------|---------|
| ALL_CONS<br>TRAINTS                  | Y      | N      | N      | N     | N      | N        | N       |
| ALL_IND_<br>PARTITIO<br>NS           | Y      | N      | N      | N     | N      | N        | N       |
| ALL_POLI<br>CIES                     | Y      | N      | N      | N     | N      | N        | N       |
| ALLOC_D<br>ETAIL                     | Y      | N      | N      | N     | N      | N        | Y       |
| ALLOC_H<br>EADER                     | Y      | N      | N      | N     | N      | N        | Y       |
| CLASS                                | Y      | N      | N      | N     | N      | N        | N       |
| CLASS_SA<br>LES_FORE<br>CAST         | N      | N      | N      | Y     | N      | Y        | N       |
| CLASS_SA<br>LES_HIST                 | N      | N      | N      | N     | Y      | N        | N       |
| CLASS_SA<br>LES_HIST_<br>MTH         | Y      | N      | N      | N     | Y      | N        | N       |
| COST_CH<br>ANGE_TRI<br>GGER_TE<br>MP | Y      | N      | N      | Y     | N      | Y        | N       |
| COST_SUS<br>P_HEAD                   | N      | N      | Y      | N     | N      | N        | N       |
| DAILY_DA<br>TA                       | Y      | N      | N      | N     | N      | N        | N       |

| Table                          | Select | Insert | Update | Index | Delete | Truncate | Trigger |
|--------------------------------|--------|--------|--------|-------|--------|----------|---------|
| DAILY_DATA_TEMP                | Y      | N      | N      | N     | N      | Y        | N       |
| DBA_INDEXES                    | Y      | N      | N      | N     | N      | N        | N       |
| DBA_TRIGGERS                   | Y      | N      | N      | N     | N      | N        | N       |
| DEALFACT_TEMP                  | N      | Y      | N      | N     | N      | N        | N       |
| DEAL_ACTUALS_FO<br>RECAST      | Y      | N      | N      | N     | N      | N        | N       |
| DEAL_ACTUALS_IT<br>EM_LOC      | Y      | Y      | N      | N     | N      | N        | N       |
| DEAL_BB_NO_REBA<br>TE_TEMP     | N      | Y      | N      | N     | N      | Y        | N       |
| DEAL_BB_REBATE_P<br>O_TEMP     | N      | Y      | N      | N     | N      | Y        | N       |
| DEAL_BB_RECEIPT_S<br>ALES_TEMP | N      | Y      | N      | N     | N      | Y        | N       |
| DEAL_HEAD                      | Y      | N      | Y      | N     | N      | N        | N       |
| DEAL_DETAIL                    | Y      | N      | N      | N     | N      | N        | N       |
| DEAL_PERF_TRAN_D<br>ATA        | Y      | N      | N      | N     | N      | N        | N       |
| DEAL_ITEM_LOC_E<br>XPLODE      | Y      | N      | N      | N     | N      | N        | N       |
| DEAL_SKU_TEMP                  | N      | N      | N      | Y     | N      | Y        | N       |
| DEAL_TRAN_DATA<br>_TEMP        | N      | Y      | N      | N     | N      | Y        | N       |
| DEPS                           | Y      | N      | N      | N     | N      | N        | N       |

| Table               | Select | Insert | Update | Index | Delete | Truncate | Trigger |
|---------------------|--------|--------|--------|-------|--------|----------|---------|
| DEPT_SALES_FORECAST | N      | N      | N      | Y     | N      | Y        | N       |
| DEPT_SALES_HIST     | N      | N      | N      | N     | Y      | N        | N       |
| DEPT_SALES_HIST_MTH | Y      | N      | N      | N     | Y      | N        | N       |
| DOMAIN_CLASS        | N      | N      | Y      | N     | N      | N        | N       |
| DOMAIN_DEPT         | N      | N      | Y      | N     | N      | N        | N       |
| DOMAIN_SUBCLASS     | N      | N      | Y      | N     | N      | N        | N       |
| EDI_DAILY_SALES     | N      | N      | N      | N     | Y      | N        | N       |
| EDI_ORDER_TEMP      | N      | N      | N      | Y     | N      | Y        | N       |
| EDI_SUPS_TEMP       | N      | Y      | N      | N     | N      | N        | N       |
| FIF_RECEIVING       | N      | Y      | N      | Y     | N      | Y        | N       |
| FIXED_DEAL          | Y      | N      | Y      | N     | N      | N        | N       |
| FORECAST_REBUILD    | N      | N      | N      | Y     | N      | Y        | N       |
| GROUPS              | Y      | N      | N      | N     | N      | N        | N       |
| HIST_REBUILD_MASK   | Y      | N      | N      | Y     | N      | Y        | N       |
| IB_RESULTS          | N      | N      | Y      | N     | N      | N        | N       |
| IF_TRANSACTION_DATA | Y      | N      | N      | N     | N      | N        | N       |
| INVC_DETAIL         | N      | N      | Y      | N     | N      | N        | N       |
| INVC_DETAIL_TEMP    | Y      | N      | N      | N     | N      | Y        | N       |

| Table                 | Select | Insert | Update | Index | Delete | Truncate | Trigger |
|-----------------------|--------|--------|--------|-------|--------|----------|---------|
| INVC_DETAIL_TEMP2     | N      | N      | N      | N     | N      | Y        | N       |
| INVC_HEADER           | N      | N      | Y      | N     | N      | N        | N       |
| INVC_HEADER_TEMP      | Y      | N      | N      | N     | N      | Y        | N       |
| ITEM_FORECAST         | N      | N      | N      | Y     | N      | N        | N       |
| ITEM_LOC              | Y      | N      | N      | N     | N      | N        | N       |
| ITEM_LOC_TEMP         | N      | Y      | N      | N     | N      | Y        | N       |
| ITEM_MASTER           | Y      | N      | N      | N     | N      | N        | N       |
| ITEM_SUPP_COUNTRY     | Y      | N      | N      | N     | N      | N        | N       |
| ITEM_SUPP_COUNTRY_LOC | Y      | N      | N      | N     | N      | N        | N       |
| MC_REJECTIONS         | N      | N      | N      | Y     | N      | Y        | N       |
| MOD_ORDER_ITEMHTS     | N      | N      | N      | Y     | N      | Y        | N       |
| ON_ORDER_TEMP         | N      | N      | N      | Y     | N      | Y        | N       |
| ORD_MISSED            | N      | N      | N      | Y     | N      | Y        | N       |
| ORD_TEMP              | N      | N      | N      | Y     | N      | Y        | N       |
| ORDHEAD               | Y      | N      | N      | N     | N      | N        | N       |
| ORDLOC                | Y      | N      | N      | N     | N      | N        | N       |
| ORDSKU                | Y      | N      | N      | N     | N      | N        | N       |
| PACKITEM              | Y      | N      | N      | N     | N      | N        | N       |
| PERIOD                | Y      | N      | N      | N     | N      | N        | N       |
| POS_BUTTON_ON_HEAD    | N      | N      | Y      | N     | N      | N        | N       |

| Table                          | Select | Insert | Update | Index | Delete | Truncate | Trigger |
|--------------------------------|--------|--------|--------|-------|--------|----------|---------|
| POS_COUPON_HEAD                | N      | N      | Y      | N     | N      | N        | N       |
| POS_MERCH_CRITERIA             | N      | N      | Y      | N     | N      | N        | N       |
| POS_MODS                       | N      | Y      | N      | Y     | N      | Y        | N       |
| POS_MONETARY_ORDER_HEAD        | N      | N      | Y      | N     | N      | N        | N       |
| POS_PAYMENT_OUT_HEAD           | N      | N      | Y      | N     | N      | N        | N       |
| POS_PRODUCT_REST_HEAD          | N      | N      | Y      | N     | N      | N        | N       |
| POS_STORE                      | N      | N      | Y      | N     | N      | N        | N       |
| POS_SUPP_PAYMENT_CRITERIA      | N      | N      | Y      | N     | N      | N        | N       |
| POS_TENDER_TYPE_HEAD           | N      | N      | Y      | N     | N      | N        | N       |
| RECLASS_COST_CHG_QUEUE         | Y      | Y      | Y      | N     | N      | N        | N       |
| RECLASS_HEAD                   | Y      | N      | N      | N     | N      | N        | N       |
| RECLASS_ITEM                   | Y      | N      | N      | N     | N      | N        | Y       |
| RECLASS_TRIGGER_TEMP           | Y      | N      | N      | Y     | Y      | Y        | N       |
| REPL_ATTRIBUTES_UPDATE_EXCLUDE | Y      | Y      | N      | N     | Y      | N        | N       |
| REPL_ATTRIBUTES_UPDATE_HEAD    | Y      | Y      | N      | N     | Y      | N        | N       |

| Table                                 | Select | Insert | Update | Index | Delete | Truncate | Trigger |
|---------------------------------------|--------|--------|--------|-------|--------|----------|---------|
| REPL_ATT<br>R_UPDATE<br>_ITEM         | Y      | Y      | Y      | N     | Y      | N        | N       |
| REPL_ATT<br>R_UPDATE<br>_LOC          | Y      | Y      | N      | N     | Y      | N        | N       |
| REPL_DAY                              | Y      | Y      | N      | N     | N      | N        | N       |
| REPL_ITE<br>M_LOC                     | Y      | Y      | N      | N     | N      | N        | N       |
| REPL_ITE<br>M_LOC_U<br>PDATES         | N      | Y      | N      | Y     | N      | N        | N       |
| RESTART_<br>CONTROL                   | Y      | N      | N      | N     | N      | N        | N       |
| RPL_ALLO<br>C_IN_TMP                  | N      | Y      | N      | N     | N      | Y        | N       |
| RPL_DIST<br>RO_TMP                    | N      | Y      | N      | N     | N      | Y        | N       |
| RTV_HEA<br>D                          | Y      | N      | N      | N     | N      | N        | N       |
| SALWEEK<br>_C_DAILY                   | N      | Y      | N      | N     | N      | Y        | N       |
| SALWEEK<br>_C_WEEK                    | Y      | Y      | N      | N     | N      | Y        | N       |
| SALWEEK<br>_RESTART<br>_DEPT          | Y      | Y      | Y      | N     | N      | Y        | N       |
| SEC_USER<br>_ZONE_M<br>ATRIX          | N      | N      | N      | Y     | N      | Y        | N       |
| STAGE_CO<br>MPLEX_D<br>EAL_DETA<br>IL | N      | N      | N      | N     | N      | Y        | N       |
| STAGE_CO<br>MPLEX_D<br>EAL_HEA<br>D   | N      | N      | N      | N     | N      | Y        | N       |

| Table                           | Select | Insert | Update | Index | Delete | Truncate | Trigger |
|---------------------------------|--------|--------|--------|-------|--------|----------|---------|
| STAGE_FI<br>XED_DEA<br>L_DETAIL | N      | N      | N      | N     | N      | Y        | N       |
| STAGE_FI<br>XED_DEA<br>L_HEAD   | N      | N      | N      | N     | N      | Y        | N       |
| STAKE_HE<br>AD                  | Y      | N      | N      | N     | N      | N        | N       |
| STAKE_PR<br>OD_LOC              | Y      | N      | N      | N     | N      | N        | N       |
| STAKE_SK<br>U_LOC               | Y      | N      | N      | N     | N      | N        | N       |
| STORE                           | Y      | N      | Y      | N     | N      | N        | N       |
| STORE_AD<br>D                   | Y      | N      | N      | N     | Y      | N        | N       |
| SUBCLASS<br>_SALES_F<br>ORECAST | N      | N      | N      | Y     | N      | N        | N       |
| SUBCLASS<br>_SALES_HI<br>ST     | N      | N      | N      | N     | Y      | N        | N       |
| SUBCLASS<br>_SALES_HI<br>ST_MTH | Y      | N      | N      | N     | Y      | N        | N       |
| SUPS                            | Y      | N      | N      | N     | N      | N        | N       |
| SUP_DATA                        | N      | N      | N      | N     | Y      | N        | N       |
| SUPS_MIN<br>_FAIL               | N      | N      | N      | Y     | N      | Y        | N       |
| SYSTEM_<br>OPTIONS              | Y      | N      | N      | N     | N      | N        | N       |
| SYSTEM_<br>VARIABLE<br>S        | Y      | N      | Y      | N     | N      | N        | N       |
| TEMP_TR<br>AN_DATA              | Y      | N      | N      | Y     | N      | Y        | N       |
| TEMP_TR<br>AN_DATA<br>_SUM      | N      | Y      | N      | Y     | N      | Y        | N       |

| Table               | Select | Insert | Update | Index | Delete | Truncate | Trigger |
|---------------------|--------|--------|--------|-------|--------|----------|---------|
| TIF_EXPL<br>ODE     | N      | N      | N      | Y     | N      | Y        | N       |
| TRAN_DA<br>TA       | N      | Y      | N      | N     | N      | N        | N       |
| TSF_HEAD            | N      | N      | Y      | N     | N      | N        | N       |
| VAT_COD<br>E_RATES  | Y      | N      | N      | N     | N      | N        | N       |
| VAT_ITEM            | Y      | N      | N      | N     | N      | N        | N       |
| VENDINV<br>C_TEMP   | N      | Y      | N      | N     | N      | Y        | N       |
| WEEK_DA<br>TA_TEMP  | N      | N      | N      | N     | N      | Y        | N       |
| WEEK_DA<br>TA       | Y      | N      | N      | N     | N      | N        | N       |
| WH                  | Y      | N      | N      | N     | N      | N        | N       |
| WH_STOR<br>E_ASSIGN | N      | N      | N      | N     | Y      | N        | N       |

**I/O Specification**

NA



## Pricing Interface to Oracle Retail Price Management (RPM)

### Overview

The batch program described in this chapter sends the smoothed average item retail data to a staging table (IF\_RPM\_SMOOTHED\_AVG) for the purposes of RPM integration.

### Batch Design Summary

The following batch design is included in this functional area:

- RPPMOVAVG (RPM Moving Average)

### rpmmovavg (RPM Moving Average)

#### Functional Area

Pricing

#### Module Affected

RPPMOVAVG.PC

#### Design Overview

This batch module takes the number of units sold from TRAN\_DATA table for all items designated for a particular store within a specified store/day, and maintains a smoothed average in the IF\_RPM\_SMOOTHED\_AVG table.

Only the sales, which have a sales type of regular, are included. If the item is on promotion or clearance, then no updating is required. The units under normal sales are considered as unadjusted units and are taken for smoothed average. The threshold percent is maintained at the department level. This percent is compared to the existing smoothed average value and used to limit the upper and lower boundaries for regular sales received. If the unadjusted units amount is outside of the boundaries, then the appropriate boundary amount is substituted and become the adjusted units amount. If no threshold percent is defined for the department, it is defaulted to 50%.

#### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                   |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (Daily)                                                                                                                                                               |
| Scheduling Considerations | This program has to be run after all of the data for the store/day is uploaded into RMS and before the days data is removed from the TRAN_DATA table. Run before SALSTAGE.PC. |
| Pre-Processing            | N/A                                                                                                                                                                           |
| Post-Processing           | N/A                                                                                                                                                                           |

| Schedule Information | Description |
|----------------------|-------------|
| Threading Scheme     | N/A         |

**Restart/Recovery**

The logical unit of work for this program is set at store/item level.

Restart ability is implied based on item and store combination. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table               | Select | Insert | Update | Delete |
|---------------------|--------|--------|--------|--------|
| ITEM_LOC            | Yes    | No     | No     | No     |
| ITEM_MASTER         | Yes    | No     | No     | No     |
| LOCATION_CLOSED     | Yes    | No     | No     | No     |
| TRAN_DATA           | Yes    | No     | No     | No     |
| DEPS                | Yes    | No     | No     | No     |
| IF_RPM_SMOOTHED_AVG | Yes    | Yes    | Yes    | No     |

**I/O Specification**

N/A

---

## Purchase Order Batch

### Overview

The batch modules described in this section run internally in RMS primarily for the purpose of maintaining PO data within the system.

### Batch Design Summary

The following batch designs are included in this functional area:

- GENPREISS.PC (Pre-issued Order Number Generation)
- ORDAUTCL.PC (Purchase Order Auto Close)
- ORDPRG.PC (Purchase Order Purge)
- ORDREV.PC (Purchase Order Information Written to Order History Tables)
- ORDUPD.PC (Retail Price Change on Purchase Orders)
- VRPLBLD.PC (Vendor Replenished Order Build)

### genpreiss (Pre-Issued Order Number Generation)

#### Functional Area

Purchase Orders

#### Module Affected

GENPREISS.PC

#### Design Overview

Based on records on the SUPP\_PREISSUE table, this batch program reserves order numbers for suppliers that do VMI by placing these pre-generated order numbers on the ORD\_PREISSUE table.

#### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                                            |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 1 or Ad Hoc                                                                                                                                                                                      |
| Scheduling Considerations | This module can be run at any stage in the batch cycle. It is independent of other programs. If a custom program is created to download the pre-issued numbers, it needs to be run after GENPREISS.PC. |
| Pre-Processing            | N/A                                                                                                                                                                                                    |
| Post-Processing           | N/A                                                                                                                                                                                                    |
| Threading Scheme          | Multi-threaded by supplier                                                                                                                                                                             |

**Restart/Recovery**

The logical unit of work for this program is set at the supplier level, based on a single record from the SUPP\_PREISSUE table. It uses v\_restart\_supplier to achieve restart/recovery.

The changes are posted when the commit\_max\_ctr value is reached and the value of the counter is subject to change based on implementation. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table         | Select | Insert | Update | Delete |
|---------------|--------|--------|--------|--------|
| SUPP_PREISSUE | Yes    | No     | Yes    | No     |
| ORD_PREISSUE  | No     | Yes    | No     | No     |

**I/O Specification**

N/A

**ordautcl (PO Auto Close)****Functional Area**

Purchase orders

**Module Affected**

ORDAUTCL.PC

**Design Overview**

This batch program is used to process POs that need to be deleted or closed that meet certain conditions. The criteria are as mentioned below:

Category 1:

- The order is not in 'C'ompleted status and was previously approved.
- The number of days between the latest ship date and the current date is greater than the 'Approved PO Close Delay' SYSTEM\_OPTIONS setting.
- There are no open shipments for the order.

Category 2:

- The order is not in 'C'ompleted status and was previously approved.
- A specified amount of time (approved PO close delay in SYSTEM\_OPTIONS) after the not after date of the PO has passed.

- A specified amount of time (partially received PO close delay in SYSTEM\_OPTIONS) after the not after date has passed.
- A specified amount of time (partially received PO close delay in SYSTEM\_OPTIONS) after the expected receipt date (or shipped date if the expected date was not captured) has passed.
- There are no open appointments in the system for the order.

#### Category 3:

- The order has a status of worksheet or submitted, and the order has never been previously approved.
- The number of days between the current date and the order creation date is greater than the 'Worksheet PO Clean Up Delay' in SYSTEM\_OPTIONS.
- The order is a manual order (not created by replenishment).

Retrieved orders are subsequently processed based on their category:

1. Category 1 orders are closed. Closing an order involves adjusting the order quantities, shipment quantities and OTB. Any allocation associated with the order also is closed if it is released 'X' number of days before vdate. The 'X' number of days is defaulted from an external system and set on the RMS codes table for code\_type 'DEFT'.
2. For category 2 orders, orders are closed if there are no pending receipts or if the 'Auto Close Partially Received' system indicator is set to 'Y'.
3. Category 3 orders are deleted from the system.

### Scheduling Constraints

| Schedule Information      | Description                                                                                     |
|---------------------------|-------------------------------------------------------------------------------------------------|
| Processing Cycle          | Ad Hoc                                                                                          |
| Scheduling Considerations | The program should be run in the final phase of the batch along with the other purging modules. |
| Pre-Processing            | N/A                                                                                             |
| Post-Processing           | N/A                                                                                             |
| Threading Scheme          | N/A                                                                                             |

### Restart/Recovery

Restart recovery is implicit since the program purges and cancels records in the database one order at a time.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

## Key Tables Affected

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| ORDHEAD                | Yes    | No     | Yes    | Yes    |
| SHIPMENT               | Yes    | No     | Yes    | No     |
| APPT_HEAD              | Yes    | No     | No     | No     |
| APPT_DETAIL            | Yes    | No     | No     | No     |
| SHIPSKU                | Yes    | No     | Yes    | No     |
| ORDLOC                 | No     | No     | Yes    | Yes    |
| ALLOC_DETAIL           | No     | No     | Yes    | Yes    |
| OBLIGATION_COMP        | No     | No     | No     | Yes    |
| WO_DETAIL              | No     | No     | No     | yes    |
| WO_HEAD                | No     | No     | No     | Yes    |
| WO_SKU_LOC             | No     | No     | No     | Yes    |
| WO_WIP                 | No     | No     | No     | Yes    |
| ALLOC_CHRG             | No     | No     | No     | Yes    |
| ALLOC_HEADER           | No     | No     | No     | Yes    |
| ORDLOC_DISCOUNT        | No     | No     | No     | Yes    |
| TIMELINE               | No     | No     | No     | Yes    |
| ORDSKU_TEMP            | No     | No     | No     | Yes    |
| ORDLOC_TEM             | No     | No     | No     | Yes    |
| ALLOC_CHRG_TEMP        | No     | No     | No     | Yes    |
| ALLOC_DETAIL_TEMP      | No     | No     | No     | Yes    |
| ALLOC_HEADER_TEMP      | No     | No     | No     | Yes    |
| ORDLOC_EXP_TEMP        | No     | No     | No     | Yes    |
| ORDSKU HTS_ASSESS_TEMP | No     | No     | No     | Yes    |
| ORDSKU HTS_TEMP        | No     | No     | No     | Yes    |
| ORDLOC_DISCOUNT_TEMP   | No     | No     | No     | Yes    |
| TIMELINE_TEMP          | No     | No     | No     | Yes    |
| REQ_DOC_TEMP           | No     | No     | No     | Yes    |
| WO_DETAIL_TEMP         | No     | No     | No     | Yes    |
| WO_HEAD_TEMP           | No     | No     | No     | Yes    |
| ORDLOC_WKSHT           | No     | No     | No     | Yes    |
| ORDLOC_REV             | No     | No     | No     | Yes    |

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| ORDSKU_REV      | No     | No     | No     | Yes    |
| ORDSKU          | No     | No     | No     | Yes    |
| ORDCUST         | No     | No     | No     | Yes    |
| ORDHEAD_REV     | No     | No     | No     | Yes    |
| ORDLC           | No     | No     | No     | Yes    |
| DEAL_COMP_PROM  | No     | No     | No     | Yes    |
| DEAL_ITEMLOC    | No     | No     | No     | Yes    |
| DEAL_THRESHOLD  | No     | No     | No     | Yes    |
| DEAL_DETAIL     | No     | No     | No     | Yes    |
| DEAL_QUEUE      | No     | No     | No     | Yes    |
| DEAL_CALC_QUEUE | No     | No     | No     | Yes    |
| DEAL_HEAD       | No     | No     | No     | Yes    |
| ORD_INV_MGMT    | No     | No     | No     | Yes    |
| REPL_RESULTS    | No     | No     | No     | Yes    |
| REV_ORDERS      | No     | No     | No     | Yes    |
| REQ_DOC         | No     | No     | No     | Yes    |
| ORD_PREISSUE    | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**ordprg (Order Purge)****Functional Area**

Purchase orders

**Module Affected**

ORDPRG.PC

**Design Overview**

The purpose of this module is to remove old orders from the system.

If the import indicator on the SYSTEM OPTIONS table (import\_ind) is 'N' and if invoice matching is not installed, then all details associated with an order are deleted when the order is closed for more months than specified in UNIT\_OPTIONS (order\_history\_months). Orders are only deleted if all allocations associated, if any, are closed. If invoice matching is installed, then all details associated with an order are deleted when the order is closed for more months than specified in UNIT\_OPTIONS (order\_history\_months). Orders are deleted only if allocations associated are closed,

shipments from the order are completely matched to invoices or closed, and all those invoices are posted.

If the import indicator on the SYSTEM OPTIONS table (import\_ind) is 'Y' and if invoice matching is not installed, then all details associated with the order are deleted when the order is closed for more months than specified in UNIT\_OPTIONS (order\_history\_months). This action presupposes that all ALC records associated with an order are in 'Processed' status, specified in ALC\_HEAD (status) and allocations associated to the order, if any, are closed.

If invoice matching is installed, then all details associated with an order are deleted when the order is closed for more months than specified in UNIT\_OPTIONS (order\_history\_months). This action presupposes that all ALC records associated with an order are in 'Processed' status, specified in ALC\_HEAD (status), all allocations associated to the order, if any, are closed, all shipments from the order are completely matched to invoices or closed, and all those invoices are posted.

This program also creates a PO header flat file to interface with the RWMS system. When orders are deleted, a record with the action type = 'D' is written to an output file. RWMS then processes this file and deletes the PO from the warehouse's database to maintain consistency between the host and warehouse environment.

### **Scheduling Constraints**

| <b>Schedule Information</b> | <b>Description</b> |
|-----------------------------|--------------------|
| Processing Cycle            | Ad Hoc (Monthly)   |
| Scheduling Considerations   | before INVPRG      |
| Pre-Processing              | N/A                |
| Post-Processing             | N/A                |
| Threading Scheme            | N/A                |

### **Restart/Recovery**

Restart ability is implied, because the records that are selected from the driving cursor is deleted before the commit. Restart library functions are still included to ensure that rollback segments are not exceeded (by committing at intervals) and to perform basic record keeping functionality.

### **Locking Strategy**

N/A

### **Security Considerations**

N/A

### **Performance Considerations**

N/A



**Key Tables Affected**

| Table               | Select | Insert | Update | Delete |
|---------------------|--------|--------|--------|--------|
| UNIT_OPTIONS        | Yes    | No     | No     | No     |
| ORDHEAD             | Yes    | No     | No     | Yes    |
| ORDLC               | Yes    | No     | No     | No     |
| ALLOC_HEADER        | Yes    | No     | No     | Yes    |
| SHIPMENT            | Yes    | No     | No     | Yes    |
| SHIPSKU             | Yes    | No     | Yes    | Yes    |
| INVC_HEAD           | Yes    | No     | No     | Yes    |
| ORDLOC_REV          | No     | No     | No     | Yes    |
| ORDHEAD_REV         | No     | No     | No     | Yes    |
| ALLOC_REV           | No     | No     | No     | Yes    |
| ALC_HEAD            | Yes    | No     | No     | Yes    |
| ALC_COMP_LOC        | No     | No     | No     | Yes    |
| OBLIGATION_COMP_LOC | No     | No     | No     | Yes    |
| OBLIGATION_COMP     | No     | No     | No     | Yes    |
| OBLIGATION          | No     | No     | No     | Yes    |
| TRANSPORTATION      | Yes    | No     | No     | Yes    |
| MISSING_DOC         | No     | No     | No     | Yes    |
| TRANS_PACKING       | No     | No     | No     | Yes    |
| TRANS_DELIVERY      | No     | No     | No     | Yes    |
| TRANS_CLAIMS        | No     | No     | No     | Yes    |
| TRANS_LIC_VISA      | No     | No     | No     | Yes    |
| TRANS_SKU           | No     | No     | No     | Yes    |
| CE_ORD_ITEM         | Yes    | No     | No     | Yes    |
| CE_LIC_VISA         | No     | No     | No     | Yes    |
| CE_CHARGES          | No     | No     | No     | Yes    |
| CE_SHIPMENT         | No     | No     | No     | Yes    |
| CE_PROTEST          | No     | No     | No     | Yes    |
| CE_FORMS            | No     | No     | No     | Yes    |
| CE_HEAD             | v      | No     | No     | Yes    |
| APPT_HEAD           | Yes    | No     | No     | Yes    |
| APPT_DETAIL         | Yes    | No     | No     | Yes    |

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| DOC_CLOSE_QUEUE   | No     | No     | No     | Yes    |
| DAILY_PURGE       | No     | Yes    | No     | No     |
| ORDSKU            | Yes    | No     | No     | Yes    |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| PACKITEM          | Yes    | No     | No     | No     |
| PACK_TMPL_HEAD    | Yes    | No     | No     | No     |
| RTV_DETAIL        | No     | No     | No     | Yes    |
| WO_DETAIL         | No     | No     | No     | Yes    |
| CARTON            | No     | No     | No     | Yes    |
| WO_HEAD           | Yes    | No     | No     | Yes    |
| ALLOC_CHRG        | No     | No     | No     | Yes    |
| ALLOC_DETAIL      | No     | No     | No     | Yes    |
| TIMELINE          | No     | No     | No     | Yes    |
| ORDLOC            | No     | No     | No     | Yes    |
| ORDLOC_DISCOUNT   | No     | No     | No     | Yes    |
| ORDLOC_EXP        | No     | No     | No     | Yes    |
| ORDSKU HTS_ASSESS | No     | No     | No     | Yes    |
| ORDSKU HTS        | No     | No     | No     | Yes    |
| REQ_DOC           | No     | No     | No     | Yes    |
| ORDSKU_REV        | No     | No     | No     | Yes    |
| ORDLOC_INVC_COST  | No     | No     | Yes    | Yes    |
| ORDCUST           | No     | No     | No     | Yes    |
| ORD_XDOCK_TEMP    | No     | No     | No     | Yes    |
| INVC_XREF         | No     | No     | No     | Yes    |
| INVC_MATCH_WKSHT  | No     | No     | No     | Yes    |
| ORDLOC_WKSHT      | No     | No     | No     | Yes    |
| SUP_VIOLATION     | No     | No     | No     | Yes    |
| REV_ORDERS        | No     | No     | No     | Yes    |
| LC_ORDAPPLY       | No     | No     | No     | Yes    |
| ORDHEAD_DISCOUNT  | No     | No     | No     | Yes    |
| RUA_RIB_INTERFACE | No     | No     | No     | Yes    |
| ORDLOC_TEMP       | No     | No     | No     | Yes    |

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| ALLOC_CHRG_TEMP        | No     | No     | No     | Yes    |
| ALLOC_DETAIL_TEMP      | No     | No     | No     | Yes    |
| ALLOC_HEADER_TEMP      | No     | No     | No     | Yes    |
| ORDSKU_TEMP            | No     | No     | No     | Yes    |
| ORDLOC_EXP_TEMP        | No     | No     | No     | Yes    |
| ORDSKU HTS_ASSESS_TEMP | No     | No     | No     | Yes    |
| ORDSKU HTS_TEMP        | No     | No     | No     | Yes    |
| ORDLOC_DISCOUNT_TEMP   | No     | No     | No     | Yes    |
| TIMELINE_TEMP          | No     | No     | No     | Yes    |
| REQ_DOC_TEMP           | No     | No     | No     | Yes    |
| WO_DETAIL_TEMP         | No     | No     | No     | Yes    |
| WO_HEAD_TEMP           | No     | No     | No     | Yes    |
| REPL_RESULTS_TEMP      | No     | No     | No     | Yes    |
| DEAL_COMP_PROM         | No     | No     | No     | Yes    |
| DEAL_HEAD              | Yes    | No     | No     | Yes    |
| DEAL_ITEMLOC           | No     | No     | No     | Yes    |
| DEAL_THRESHOLD         | No     | No     | No     | Yes    |
| DEAL_DETAIL            | No     | No     | No     | Yes    |
| DEAL_QUEUE             | No     | No     | No     | Yes    |
| DEAL_CALC_QUEUE        | No     | No     | No     | Yes    |
| ORD_INV_MGMT           | No     | No     | No     | Yes    |
| REPL_RESULTS           | No     | No     | No     | Yes    |
| INVC_DETAIL            | No     | No     | No     | Yes    |
| INVC_NON_MERCH         | No     | No     | No     | Yes    |
| INVC_MERCH_VAT         | No     | No     | No     | Yes    |
| INVC_DETAIL_VAT        | No     | No     | No     | Yes    |
| INVC_DISCOUNT          | No     | No     | No     | Yes    |
| INVC_TOLERANCE         | No     | No     | No     | Yes    |
| INVC_MATCH_QUEUE       | No     | No     | No     | Yes    |

### I/O Specification

N/A

## ordrev (Purchase Order Information Written to Order History Table)

### Functional Area

Purchase orders

### Module Affected

ORDREV.PC

### Design Overview

ORDREV writes versions of approved orders to order revision history tables. When orders are approved or when approved orders are modified, this program selects order numbers from the REV\_ORDERS table and writes current order information to the order/allocation revision tables. After the new version is written to the order revision tables, all records are deleted from the REV\_ORDERS table for that order\_no.

This program processes order changes made by the client that may need to be sent to the vendor. The order changes should always be referred to as 'versions' and kept clearly distinct from order 'revisions' which are vendor changes uploaded via the ediupack program.

If an order is not in approved status at the time the batch program runs, then none of the above processing occurs. The records stay on the REV\_ORDERS table until the PO is approved or deleted.

### Scheduling Constraints

| Schedule Information      | Description                         |
|---------------------------|-------------------------------------|
| Processing Cycle          | Phase 4 (Daily)                     |
| Scheduling Considerations | After ORDDSCNT and before EDIDLORD. |
| Pre-Processing            | N/A                                 |
| Post-Processing           | N/A                                 |
| Threading Scheme          | Multithreading based on order_no.   |

### Restart/Recovery

Restart ability is implied because the records that are selected from the driving cursor are deleted before the commit. Restart library functions are still included to ensure that rollback segments are not exceeded (by committing at intervals) and to perform basic record keeping functionality. The logical unit of work is order\_no.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table        | Select | Insert | Update | Delete |
|--------------|--------|--------|--------|--------|
| REV_ORDERS   | Yes    | No     | No     | Yes    |
| ORDHEAD      | Yes    | No     | Yes    | No     |
| SUPS         | Yes    | No     | No     | No     |
| ORDHEAD_REV  | Yes    | Yes    | No     | No     |
| ORDSKU       | Yes    | No     | No     | No     |
| ORDLOC       | Yes    | No     | No     | No     |
| ALLOC_HEADER | Yes    | No     | No     | No     |
| ALLOC_DETAIL | Yes    | No     | No     | No     |
| ORDSKU_REV   | No     | Yes    | No     | No     |
| ORDLOC_REV   | No     | Yes    | No     | No     |
| ALLOC_REV    | No     | Yes    | No     | No     |
| FIF_ORDHEAD  | No     | Yes    | No     | No     |

**I/O Specification**

N/A

**ordupd (Order Update)****Functional Area**

Purchase orders

**Module Affected**

ORDUPD.PC

**Design Overview**

This program is used to automatically change all retail costs on purchase orders when a retail price change is implemented for an item on the order with the status of 'Worksheet', 'Submit' and 'Approve'.

Open to buy is updated to give a more accurate picture of the retail value of open orders if the order is 'Approved' and if the department calculate the OTB as retail.

For more information on cost changes, see the chapter "Cost Change Batch" in this volume of the RMS Operations Guide. For more information on complex deals, see the chapter "Deals Maintenance Batch" in this volume of the RMS Operations Guide.

**Scheduling Constraints**

| Schedule Information | Description    |
|----------------------|----------------|
| Processing Cycle     | Phase 4(Daily) |

| Schedule Information      | Description                                                                                                                            |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| Scheduling Considerations | This program should be run after RPM price change extraction process to ensure that all price changes are handled by batch processing. |
| Pre-Processing            | N/A                                                                                                                                    |
| Post-Processing           | N/A                                                                                                                                    |
| Threading Scheme          | Multithreaded on Location                                                                                                              |

**Restart/Recovery**

This program does not contain restart/recovery logic.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table         | Select | Insert | Update | Delete |
|---------------|--------|--------|--------|--------|
| ORDLOC        | Yes    | No     | Yes    | No     |
| ORDHEAD       | Yes    | No     | No     | No     |
| PRICE_HIST    | Yes    | No     | No     | No     |
| OTB           | Yes    | No     | Yes    | No     |
| ITEM_MASTER   | Yes    | No     | No     | No     |
| DEPS          | Yes    | No     | No     | No     |
| V_PACKSKU_QTY | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**vrplbld (Vendor replenished order build)****Functional Area**

Purchase order.

**Module Affected**

VRPLBLD.PC

## Design Overview

This purpose of this module is to continue the process started by the batch program ediupack.pc of building Oracle Retail orders that reflect the vendor-generated orders as received through the EDI 855.

For more information on replenishment, see the chapter “Replenishment Batch” in this volume of the RMS Operations Guide.

## Scheduling Constraints

| Schedule Information      | Description                                         |
|---------------------------|-----------------------------------------------------|
| Processing Cycle          | Phase 3                                             |
| Scheduling Considerations | run after ediupack.pc                               |
| Pre-Processing            | N/A                                                 |
| Post-Processing           | prepost vrplbld post - truncates EDI_ORD_TEMP table |
| Threading Scheme          | Threaded by supplier.                               |

## Restart/Recovery

The logical unit of work for the program is a vendor order number, department and supplier combination. The program’s restartability is dependent on the value of the dept\_level\_orders column on the UNIT\_OPTIONS. Allowing multi-department orders (‘N’) stores restart the program from the last successfully processed vendor order number and supplier. If the system requires a department on the orders (‘Y’), then the program restarts from the last successfully processed vendor order number, department, and supplier.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| ITEM_MASTER           | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | Yes    | No     | No     | No     |
| SUP_IMPORT_ATTR       | Yes    | No     | No     | No     |
| SUPS                  | Yes    | No     | No     | No     |
| EDI_ORD_TEMP          | Yes    | No     | No     | No     |

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| WH              | Yes    | No     | No     | No     |
| ORDSKU          | Yes    | Yes    | Yes    | No     |
| ORDHEAD         | Yes    | Yes    | Yes    | No     |
| ORDLOC          | No     | Yes    | No     | No     |
| DEAL_CALC_QUEUE | Yes    | Yes    | Yes    | No     |
| PERIOD          | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS  | Yes    | No     | No     | No     |
| UNIT_OPTIONS    | Yes    | No     | No     | No     |

**I/O Specification**

N/A



---

## Reclassification Batch

### Overview

Item reclassification is the process through which an item or item list is moved from one department/class/subclass to another.

For a general discussion of merchandise hierarchy, see the chapter “Merchandise Hierarchy Subscription (External)” in volume 2 of this RMS Operations Guide.

When an item is reclassified, stock ledger transactions are written to move the inventory amount associated with this item from the old merchandise hierarchy level to the new one in the stock ledger. If there are active orders for this item, OTB is also updated.

Pos\_mods records are written for downloading to stores. History, such as sales history, is NOT moved.

### Batch Design Summary

The following batch designs are included in this functional area:

- CREMHIERDLY.PC (Create Merchandise Hierarchy Daily)
- RECLSDLY.PC (Reclass Daily)

### cremhierdly (Create Merchandise Hierarchy Daily)

#### Functional Area

Reclassification

#### Module Affected

CREMHIERDLY.PC

#### Design Overview

The CREMHIERDLY.PC batch program reads merchandise hierarchy records from the PEND\_MERCH\_HIER table whose effective date is on or prior to tomorrow. Each record is evaluated for either addition or modification to the hierarchy tables (DIVISION, GROUPS, DEPS, CLASS and SUBCLASS) based on the action and hierarchy types. The inserted/updated records are deleted from the PEND\_MERCH\_HIER table after they are successfully processed.

Note that CREMHIERDLY.PC is only necessary if RMS is *not* the system of record. The table that CREMHIERDLY.PC reads from (PEND\_MERCH\_HIER) is populated by a reclassification subscription API that is used to synch merchandise hierarchy with an external system.

#### Scheduling Constraints

| Schedule Information | Description     |
|----------------------|-----------------|
| Processing Cycle     | Phase 4 (daily) |

| Schedule Information      | Description                                                     |
|---------------------------|-----------------------------------------------------------------|
| Scheduling Considerations | The CREMHIERDLY.PC batch program must run prior to RECLSDLY.PC. |
| Pre-Processing            | N/A                                                             |
| Post-Processing           | N/A                                                             |
| Threading Scheme          | N/A                                                             |

**Restart/Recovery**

The logical unit of work for this program is set at the hier\_type, action\_type, and merch\_hier\_id level.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| PERIOD               | Yes    | No     | No     | No     |
| PEND_MERCH_HIER      | Yes    | No     | No     | Yes    |
| DOMAIN_DEPT          | Yes    | Yes    | Yes    | No     |
| DOMAIN_CLASS         | Yes    | Yes    | Yes    | No     |
| DOMAIN_SUBCLASS      | Yes    | Yes    | Yes    | No     |
| DIVISION             | No     | Yes    | Yes    | No     |
| GROUPS               | No     | Yes    | Yes    | No     |
| DEPS                 | No     | Yes    | Yes    | No     |
| CLASS                | No     | Yes    | Yes    | No     |
| SUBCLASS             | No     | Yes    | Yes    | No     |
| STOCK_LEDGER_INSERTS | No     | Yes    | No     | No     |

**I/O Specification**

N/A

## reclsdly (Reclassification of Item)

### Functional Area

Reclassification

### Module Affected

RECLSDLY.PC

### Design Overview

This batch program is executed in order to reclassify items from one merchandise hierarchy to another. The reclassification of items from one department/class and subclass to other is initiated by the RMS application online. The batch program picks up the records that should be reclassified for vdate + 1 and processes them.

### Scheduling Constraints

| Schedule Information      | Description                                       |
|---------------------------|---------------------------------------------------|
| Processing Cycle          | Phase 4 (daily)                                   |
| Scheduling Considerations | CREMHIERDLY.PC should be run prior to this batch. |
| Pre-Processing            | Prepost pre reclsdly                              |
| Post-Processing           | N/A                                               |
| Threading Scheme          | Threaded by reclass_no                            |

### Restart/Recovery

The logical unit of work is the combination of reclass\_no and item. Restart ability is also based on reclass\_no and item.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table        | Select | Insert | Update | Delete |
|--------------|--------|--------|--------|--------|
| RECLASS_ITEM | Yes    | No     | No     | Yes    |
| RECLASS_HEAD | Yes    | No     | No     | Yes    |
| ITEM_MASTER  | Yes    | No     | Yes    | No     |
| DEPS         | Yes    | No     | No     | No     |

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| GROUPS                | Yes    | No     | No     | No     |
| PACKITEM              | Yes    | No     | No     | No     |
| DEAL_ITEM_LOC_EXPLODE | Yes    | No     | No     | Yes    |
| DEAL_ITEMLOC          | Yes    | No     | No     | No     |
| DEAL_HEAD             | Yes    | No     | No     | No     |
| ORDHEAD               | Yes    | No     | Yes    | No     |
| ORDSKU                | Yes    | No     | No     | No     |
| DEAL_CALC_QUEUE       | Yes    | Yes    | No     | No     |
| HIST_REBUILD_MASK     | No     | Yes    | No     | No     |
| RECLASS_ERROR_LOG     | No     | Yes    | Yes    | Yes    |
| STAKE_SKU_LOC         | Yes    | Yes    | Yes    | Yes    |
| ITEM_LOC_SOH          | Yes    | No     | Yes    | No     |
| REPL_ITEM_LOC_UPDATES | No     | Yes    | No     | No     |
| POS_MODS              | No     | Yes    | No     | No     |
| TRAN_DATA             | No     | Yes    | No     | No     |
| OTB                   | Yes    | Yes    | Yes    | No     |
| SKULIST_DEPT          | Yes    | Yes    | No     | No     |
| MC_REJECTIONS         | No     | Yes    | No     | No     |

**I/O Specification**

NA

---

## Replenishment Batch

### Overview

Replenishment batch module components are designed to manage stock levels, by using stock order allocations. Only RMS replenishment functionality and Oracle Retail Allocation can create stock order allocations. This overview describes batch functionality for replenishment, including investment buy, along with descriptions of the major tables involved in the replenishment process.

### Replenishment Process

Replenishment operates in this sequence:

1. Build the purchase order
2. Scale the order
3. Split the order among trucks
4. Compare approved replenishment orders against applicable vendor minimums and reset back to 'Worksheet' status those orders that do not meet minimum quantities

### Code Values

The modules REQEXT.PC and RPLEXT.PC use code values to calculate the recommended order quantities for the item-location. Those code values are located on the REPL\_ITEM\_LOC table. Replenishment method values include the following:

- C – Constant
- M – Minimum/Maximum
- F – Floating Point
- T – Time Supply (used with forecasting)
- T – Time Supply Seasonal (used with forecasting)
- TI – Time Supply – Issues (used with forecasting)
- D – Dynamic (used with forecasting)
- D – Dynamic Seasonal (used with forecasting)
- DI – Dynamic – Issues (used with forecasting)
- SO – Store Orders.

### Investment Buy

Investment buy facilitates the process of purchasing inventory in excess of the replenishment recommendation in order to take advantage of a supplier deal or to leverage inventory against a cost increase. The inventory is stored at the warehouse or in outside storage to be used for future issues to the stores. The recommended quantity to 'investment buy', that is, to order, is calculated based on the following:

- Amount of the deal or cost increase
- Upcoming deals for the product
- Cost of money

- Cost of storage
- Forecasted demand for the product, using warehouse issue values calculated by Oracle Retail Demand Forecasting
- Target return on investment (ROI)

The rationale is to purchase as much product as profitable at the lower cost and to retain this profit rather than passing the discount on to customers and stores. The determination of how much product is profitable to purchase is based on the cost savings of the product versus the costs to purchase, store and handle the additional inventory.

Investment buy eligibility and order control are set at one of these four levels:

- Supplier
- Supplier-department
- Supplier-location (warehouse locations only)
- Supplier-department-location

Warehouses must be enabled for both replenishment and investment buy on RMS' WH (warehouse) table. In a multi-channel environment, virtual warehouses are linked to the physical warehouse.

The investment buy opportunity calculation takes place nightly during the batch run, after the replenishment need determination, but before the replenishment order build. The investment buy module IBCALC.PC attempts to purchase additional inventory beyond the replenishment recommendation in order to achieve future cost savings. Two distinct events provide the incentive to purchase investment buy quantities:

- A current supplier deal ends within the look-ahead period.
- A future cost increase becomes active within the look-ahead period.

The calculation determines the future cost for a given item-supplier-country-location for physical warehouse locations only.

If the order control for a particular line item is 'buyer worksheet', it may be modified in the buyer worksheet dialog, and can be added to either new or existing purchase orders.

## Investment Buy System Options

The following columns are held on the SYSTEM\_OPTIONS table for investment buy:

- look\_ahead\_days—The number of days before a cost event (end of a deal, or a cost increase) that the investment buy opportunity begins to calculate an event
- cost\_wh\_storage—Contains the default cost of warehouse storage, expressed as the weekly cost based on the unit of measure specified in this table's COST\_WH\_STORAGE\_UOM column. This value is held in the primary system currency. You can change this value at the warehouse or warehouse-department level.
- cost\_out\_storage—Contains the default cost of outside storage, expressed as the weekly cost base on the unit of measure specified in COST\_OUT\_STORAGE\_UOM. This value is held in the primary system currency. You can change this value at the warehouse or warehouse-department level.
- cost\_level—Indicates which cost bucket is used when calculating the return on investment for investment buy opportunities. Valid values are 'N' for net cost, 'NN' for net net cost and 'DNN' for dead net net cost.
- storage\_type—Indicates which type of storage cost should be used as the default storage cost when calculating investment buy opportunities. Valid values are

'Warehouse and 'Outside. You can change this value at the warehouse or warehouse-department level.

- `max_weeks_supply`—Contains the default maximum weeks of supply to use in the investment buy opportunity calculation. The calculation does not recommend an order quantity that would stock the associated location (currently warehouses only) for a period beyond this number of weeks. You can change this value at the warehouse or warehouse-department level.
- `target_roi`—Contains the default return on investment that must be met or exceeded for the investment buy opportunity to recommend an order quantity. You can change this value at the warehouse or warehouse-department level.
- `ib_results_purge_days`—Contains the number of days that records on the investment buy results table (IB\_RESULTS) should be kept before being purged. If an investment buy result record's `create_date` plus this value is equal to or beyond the current system date, the record is deleted by the PREPOST batch module prior to the investment buy opportunity calculation.

---

**Note:** See also the RMS Data Model for a complete description of the SYSTEM\_OPTIONS table and the investment buy columns.

---

## Batch Design Summary

The following batch designs are included in this functional area:

- CNTRPRSS.PC (Contract Replenishment Processing)
- IBCALC.PC (Investment Buy Calculation)
- IBEXPL.PC (Investment Buy Explode)
- OCIROQ (Recommended Order Quantity)
- REPLADJ.PC (Replenishment Adjustment)
- REQEXT.PC (Replenishment Quantity Extract)
- RILMAINT.PC (Replenishment Item Location Maintenance)
- RPLAPPRV.PC (Replenishment Approve)
- RPLATUPD.PC (Replenishment Attribute Update)
- RPLBLD.PC (Replenishment Order Build)
- RPLEXT.PC (Replenishment Extract)
- RPLPRG.PC (Replenishment Purge)
- RPLPRG\_MONTH (Replenishment Purge Monthly)
- RPLSPLIT.PC (Replenishment Splitting)
- SUPCNSTR.PC (Supplier Constraint Scaling)

## cntrprss (Contract Replenishment)

### Functional Area

Contracts

### Module Affected

CNTRPRSS.PC

## Design Overview

This module evaluates contracts of type A, C, and D. Contracts are ranked so that orders are created off the best contracts first. The criteria for ranking are smallest lead-time, cheapest cost, contract status (closed preferred over open), and contract type (type C are preferred over D).

It updates the temporary orders created by the item replenishment extract (RPLEXT.PC) module with the contract and supplier information of the best available contract for each item and populates the repl\_results table. This module is only run if contracting is turned on in the system.

## Scheduling Constraints

| Schedule Information      | Description                                 |
|---------------------------|---------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                             |
| Scheduling Considerations | Must be run after RPLEXT and before RPLBLD. |
| Pre-Processing            | N/A                                         |
| Post-Processing           | N/A                                         |
| Threading Scheme          | This module is threaded by dept.            |

## Restart/Recovery

As the item requirements can span across different locations, the logical unit of work varies for each item requirement. For each item requirements, records are committed to the database.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| ORD_TEMP        | Yes    | Yes    | Yes    | Yes    |
| REPL_RESULTS    | Yes    | No     | Yes    | No     |
| PERIOD          | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS  | Yes    | No     | No     | No     |
| CONTRACT_DETAIL | Yes    | No     | Yes    | No     |
| CONTRACT_HEADER | Yes    | No     | Yes    | No     |



| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| CONTRACT_COST     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY | Yes    | No     | No     | No     |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| SUPS              | Yes    | No     | No     | No     |
| ORD_MISSED        | No     | Yes    | No     | No     |
| SUP_AVAIL         | Yes    | No     | Yes    | No     |

**I/O Specification**

N/A

**ibcalc (Investment Buy Calculation)****Functional Area**

Investment Buy

**Module Affected**

IBCASC.PC

**Design Overview**

The IBCASC.PC batch program is the engine of investment buy processing. It identifies investment (IB) buy opportunities and calculates recommended order quantities (ROQs) that meet the target return-on-investment (ROI)

This module stores calculate forward buy opportunities using:

- Carrying costs
- Ordering parameters
- Deals
- Cost changes
- Forecasts
- Inventory levels
- Target ROI (return on investment)

The deals and cost change components are contained on a FUTURE\_COST table. This table holds a tuple for each future date that has a costing event (for example, a cost change, deal activation/deactivation). Oracle Retail assumes default costing bracket and default deal thresholds.

**Scheduling Constraints**

| Schedule Information      | Description                                         |
|---------------------------|-----------------------------------------------------|
| Processing Cycle          | Phase 3 (Daily)                                     |
| Scheduling Considerations | After RPLEXT.PC and IBEXPL.PC.<br>Before RPLBLD.PC. |

| Schedule Information | Description                                                                           |
|----------------------|---------------------------------------------------------------------------------------|
| Pre-Processing       | Prepost ibexpl pre – set ib_results.status from 'W' (worksheet) to 'U' (unprocessed). |
| Post-Processing      | N/A                                                                                   |
| Threading Scheme     | N/A                                                                                   |

### Restart/Recovery

The logical unit of work is item and location combination.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| FUTURE_COST           | Yes    | No     | No     | No     |
| SIM_EXPL              | Yes    | No     | No     | No     |
| ITEM_LOC              | Yes    | No     | No     | No     |
| ITEM_LOC_TRAITS       | Yes    | No     | No     | No     |
| REPL_ITEM_LOC         | Yes    | No     | No     | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| PACKITEM              | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_DIM | Yes    | No     | No     | No     |
| SUPS                  | Yes    | No     | No     | No     |
| SUB_ITEMS_DETAIL      | Yes    | No     | No     | No     |
| SUB_ITEMS_HEAD        | Yes    | No     | No     | No     |
| UOM_CONVERSION        | Yes    | No     | No     | No     |
| WH                    | Yes    | No     | No     | No     |
| IB_RESULTS            | No     | Yes    | No     | No     |

**I/O Specification**

N/A

**ibexpl (Investment Buy Explosion)****Functional Area**

Investment Buy

**Module Affected**

IBEXPL.PC

**Design Overview**

The IBEXPL batch program pre-qualifies investment buy (IB) legible wh/dept and IB legible supp/dept/locs.

The WH\_DEPT table holds IB parameters at the WH or at the wh/dept level. If there are IB parameters defined at the wh/dept level, they are used. If there are not IB parameters defined at the wh/dept level, the IB parameters at the WH level are used. If IB parameters are not defined at either level, system level IB parameters (system\_options) are used. The first part of this program sends IB parameters to the wh/dept level no matter what level they are held at in the database. The results are written to the WH\_DEPT\_EXPL table.

Next wh\_dept\_expl (with all legible wh/dept combos) is combined with sup\_inv\_mgmt to get the final list of all legible sup/dept/locs.

sup\_inv\_mgmt determines whether or not a given sup/dept/loc combo is IB legible. The main problem is that this table can store information at different levels depending upon the supplier's inv\_mgmt\_lvl (SUPS.INV\_MGMT\_LVL).

- Sup (S)
- Sup/dept (D)
- Sup/loc (L)
- Sup/dept/loc (A)

If the record is not found at the level suggested by the SUPS.INV\_MGMT\_LVL, it needs to look up the hierarchy as shown below, up to the highest level (sup). If no record exists as the sup level, it is not IB legible.

- Sup
- Sup/dept -> sup
- Sup/loc -> sup
- Sup/dept/loc -> sup/dept -> sup

The second part of this program explodes the SUP\_INV\_MGMT table down to the sup/dept/loc level by filling in the implied rows. The exploded sup\_inv\_mgmt information is only done for IB legible wh/dept combinations from wh\_dept\_expl. The results are placed on sim\_expl.

**Scheduling Constraints**

| Schedule Information | Description     |
|----------------------|-----------------|
| Processing Cycle     | Phase 3 (daily) |

| Schedule Information      | Description                           |
|---------------------------|---------------------------------------|
| Scheduling Considerations | After RPTEXT.PC and before IBCALC.PC. |
| Pre-Processing            | N/A                                   |
| Post-Processing           | N/A                                   |
| Threading Scheme          | N/A                                   |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

This program requires special permissions. It must be run by an Oracle user that is granted the following privileges, or be run by a database administrator.

- ‘drop any table’

AND

- ‘alter any index’

**Performance Considerations**

N/A

**Key Tables Affected**

| Table        | Select | Insert | Update | Delete |
|--------------|--------|--------|--------|--------|
| WH           | Yes    | No     | No     | No     |
| DEPS         | Yes    | No     | No     | No     |
| WH_DEPT      | Yes    | No     | No     | No     |
| SUP_INV_MGMT | Yes    | No     | No     | No     |
| SUPS         | Yes    | No     | No     | No     |
| WH_DEPT_EXPL | Yes    | Yes    | No     | Yes    |
| TERMS        | Yes    | No     | No     | No     |
| SIM_EXPL     | No     | Yes    | No     | Yes    |

**I/O Specification**

N/A

## ociroq (Recommended Order Quantity)

### Functional Area

Replenishment

### Module Affected

OCIROQ.C

### Design Overview

This batch module is used to calculate the net inventory position of the items and determines the ROQ, which are on replenishment. The recommended order quantities are stored in RPL\_NET\_INVENTORY\_TMP table. This information is extracted by REQEXT (item requisition extraction).

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                                                  |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                                                                                                                                                              |
| Scheduling Considerations | RPLATUPD, RILMAINT and REPLADJ need to run before OCIROQ so that all replenishment calculation attributes are up to date.<br>POSUPLD needs to run before REQEXT so that all stock information is up to date. |
| Pre-Processing            | Prepost ociroq pre – truncate records in RPL_NET_INVENTORY_TMP tables and build RPL_DISTRO_TMP and RPL_ALLOC_IN_TMP tables.                                                                                  |
| Post-Processing           | N/A                                                                                                                                                                                                          |
| Threading Scheme          | POSIX threads.<br>The restart_control.num_threads controls the number of POSIX threads that are run within OCIROQ.<br>The batch program OCIROQ.C itself is run with one thread.                              |

### Restart/Recovery

The program processes all items on REPL\_DAY for the current day. If the program fails, the RPL\_NET\_INVENTORY\_TMP table is truncated prior to restarting.

### Locking Strategy

STORE\_ORDER table records are locked while calculating ROQ.

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| DOMAIN_CLASS          | Y      | N      | N      | N      |
| DOMAIN_DEPT           | Y      | N      | N      | N      |
| DOMAIN_SUBCLASS       | Y      | N      | N      | N      |
| REPL_DAY              | Y      | N      | N      | N      |
| REPL_ITEM_LOC         | Y      | N      | N      | N      |
| RPL_NET_INVENTORY_TMP | N      | Y      | N      | N      |
| STORE                 | Y      | N      | N      | N      |
| WH                    | Y      | N      | N      | N      |
| STORE_ORDER           | Y      | N      | Y      | N      |
| SUPS                  | Y      | N      | N      | N      |

**I/O Specification**

N/A

**repladj (Replenishment Adjustment)****Functional Area**

Replenishment

**Module Affected**

REPLADJ.PC

**Design Overview**

This batch module recalculates the maximum stock levels for all item-location combinations with replenishment method of 'F' (floating point). The floating model stock method dynamically calculates an order-up-to-level. The calculated order-up-to-level is written to REPL\_ITEM\_LOC table.

The maximum model stock (used for calculating order-up-to-level) is derived using the sales history of various periods of time in order to accommodate seasonality as well as trend. The sales history is obtained from the ITEM\_LOC\_HIST table.

**Scheduling Constraints**

| Schedule Information      | Description                                  |
|---------------------------|----------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                              |
| Scheduling Considerations | Run before RPLEXT/REQEXT and after RPLATUPD. |
| Pre-Processing            | N/A                                          |
| Post-Processing           | N/A                                          |

| Schedule Information | Description            |
|----------------------|------------------------|
| Threading Scheme     | Multi-threaded by dept |

**Restart/Recovery**

The module has restart/recovery based on item/ location. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| REPL_ITEM_LOC     | Yes    | No     | Yes    | No     |
| SUB_ITEMS_HEAD    | Yes    | No     | No     | No     |
| SUB_ITEMS_DETAIL  | Yes    | No     | No     | No     |
| ITEM_LOC_HIST     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY | Yes    | No     | No     | No     |
| REPL_DAY          | Yes    | No     | No     | No     |
| STORE             | Yes    | No     | No     | No     |
| WH                | Yes    | No     | No     | No     |
| SUPS              | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**reqext (Replenishment Quantity Extract)****Functional Area**

Replenishment

**Module Affected**

REQEXT.PC

## Design Overview

The item requisition extraction module performs the automatic replenishment of items from warehouses to stores. It runs through every item-store combination set to be reviewed on the current day, and calculates the quantity of the item, known as the recommended order quantity (ROQ) that needs to be transferred to the store (if any). In addition, it distributes this ROQ over any applicable alternate items associated with the item.

Once the transfer quantity of an item is calculated, transfers are created and records are written to the replenishment results table (REPL\_RESULTS) based on the replenishment order control indicator.

## Scheduling Constraints

| Schedule Information      | Description                                                                                                                           |
|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                                                                                       |
| Scheduling Considerations | POSUPLD, RPLATUPD, REPLADJ, PREPOST OCIROQ and OCIROQ need to run before REQEXT. RPLEXT should run after REQEXT.                      |
| Pre-Processing            | N/A                                                                                                                                   |
| Post-Processing           | PREPOST REQEXT POST – update transfer status to approved.                                                                             |
| Threading Scheme          | Multiple processes of this program can be run at the same time, each running against a different partition of repl_net_inventory_tmp. |

## Restart/Recovery

The logical unit of work is an item/source warehouse. Restart/recovery is achieved implicitly because repl\_item\_loc records that are processed are updated with a last review date and only records that are not reviewed today is picked up by the driving cursor again. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table    | Select | Insert | Update | Delete |
|----------|--------|--------|--------|--------|
| ITEM_LOC | Yes    | No     | No     | No     |



| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| ITEM_LOC_SOH          | No     | No     | Yes    | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| PACKHEAD              | Yes    | No     | No     | No     |
| PACKITEM              | Yes    | No     | No     | No     |
| PACKSTORE_HIST        | Yes    | No     | No     | No     |
| PERIOD                | Yes    | No     | No     | No     |
| RAG_SKUS_ST_HIST      | Yes    | No     | No     | No     |
| REPL_DAY              | Yes    | No     | No     | No     |
| REPL_ITEM_LOC         | Yes    | No     | Yes    | No     |
| REPL_RESULTS          | No     | Yes    | No     | No     |
| RPL_NET_INVENTORY_TMP | Yes    | No     | No     | No     |
| STORE                 | Yes    | No     | No     | No     |
| SUB_ITEMS_DETAIL      | Yes    | No     | No     | No     |
| SUB_ITEMS_HEAD        | Yes    | No     | No     | No     |
| SUPS                  | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS        | Yes    | No     | No     | No     |
| TSFDETAIL             | Yes    | Yes    | Yes    | No     |
| TSFHEAD               | Yes    | Yes    | No     | No     |
| WH                    | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**rilmaint (Replenishment Item Location Maintenance)****Functional Area**

Replenishment

**Module Affected**

RILMAINT.PC

**Design Overview**

This module transfers the replenishment attributes from REPL\_ITEM\_LOC\_UPDATES to REPL\_ITEM\_LOC. REPL\_ITEM\_LOC\_UPDATES is populated when certain attributes effecting replenishment are modified. These attributes are located across the entire system and are monitored for changes by a series of triggers and modules. Once a

change is logged in REPL\_ITEM\_LOC\_UPDATES, RILMAINT.PC notes the type of change and updates REPL\_ITEM\_LOC appropriately.

### Scheduling Constraints

| Schedule Information      | Description                                                             |
|---------------------------|-------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                         |
| Scheduling Considerations | Run after STOREADD.PC and RPLATUPD.PC but before REPLADJ.PC.            |
| Pre-Processing            | N/A                                                                     |
| Post-Processing           | prepost rilmaint post- truncate records on REPL_ITEM_LOC_UPDATES table. |
| Threading Scheme          | Threaded by location (store and warehouse)                              |

### Restart/Recovery

The logical unit of work for RILMAINT is item, change type and location. Records are committed to the database once commit\_max\_counter defined in the RESTART\_CONTROL table is reached.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| REPL_ITEM_LOC_UPDATES | Yes    | No     | No     | No     |
| REPL_ITEM_LOC         | Yes    | No     | Yes    | Yes    |
| REPL_DAY              | Yes    | No     | No     | Yes    |
| STORE_ORDERS          | No     | No     | No     | Yes    |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| PACKITEM              | Yes    | No     | No     | No     |
| ITEM_LOC              | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | Yes    | No     | No     | No     |

### I/O Specification

N/A

## rplapprv (Automatic Replenishment Order Approval)

### Functional Area

Replenishment

### Module Affected

RPLAPPRV.PC

### Design Overview

This program looks at all replenishment, vendor and contract orders created during the nightly batch run. These orders are compared with any vendor minimums that may exist. Orders that do not meet the vendor minimums are either deleted or placed in worksheet status. A flag, held at the supplier inventory management level (ORD\_INV\_MGMT.ORD\_PURGE\_IND), determines what action is taken on orders that fail minimums. Vendor generated orders are not subject to these minimum checks.

Vendor minimums can be held at the order, item, or location level. Order and location level minimums are held on the SUP\_INV\_MGMT table. There is a flag that determines if they are applied at the order level or at the location level. Vendor minimums at the SKU level are held on the ITEM\_SUPP\_COUNTRY table.

When the ORD\_INV\_MGMT.ORD\_PURGE\_IND is 'N', a failure at any level causes the order to be placed in worksheet status. When the ORD\_INV\_MGMT.ORD\_PURGE\_IND is 'Y', a failure at the location level causes the offending location to be deleted; a failure at the SKU level causes the problematic SKU to be deleted; and a failure at the order level caused the entire order to be deleted.

For any orders that fail vendor minimums when the ORD\_INV\_MGMT.ORD\_PURGE\_IND is 'Y', a record is written to the SUPS\_MIN\_FAIL table for reporting purposes. This table is purged during the pre-processing of this batch program.

After order records are updated, any applicable deals, brackets and allowances are applied to the orders. Open to buy is then updated for any orders built in approved status. If any orders are contract orders, the contract amounts are updated as well to reflect any order record deletions.

This program runs in both (multi-channel and non multi-channel) environments.

An order may not pass vendor minimum checks assuming that the vendor minimum checks are performed for a physical WH. If the vendor minimum is not met for a physical location, all the virtual WHs on the order within the physical WH stores need to be removed along with associated allocations.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                              |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                                                                                                                          |
| Scheduling Considerations | This program should run directly after the replenishment SUPCNSTR program. It is important that this program runs before any other process affects the generated orders. |

| Schedule Information | Description                                          |
|----------------------|------------------------------------------------------|
| Pre-Processing       | Prepost rplapprv pre – truncates sups_min_fail table |
| Post-Processing      | N/A                                                  |
| Threading Scheme     | N/A                                                  |

### Restart/Recovery

The logical unit of work is order number. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| ORDHEAD_LOCK      | No     | No     | No     | Yes    |
| ORDHEAD           | Yes    | No     | Yes    | Yes    |
| ORDLOC            | Yes    | No     | No     | Yes    |
| ORDSKU            | Yes    | No     | No     | Yes    |
| ORD_INV_MGMT      | Yes    | No     | Yes    | Yes    |
| DEAL_CALC_QUEUE   | No     | Yes    | Yes    | No     |
| ITEM_SUPP_COUNTRY | Yes    | No     | No     | No     |
| SUPS_MIN_FAIL     | No     | Yes    | No     | Yes    |
| ALLOC_HEADER      | Yes    | No     | Yes    | Yes    |
| ALLOC_DETAIL      | No     | No     | No     | Yes    |
| CONTRACT_HEADER   | Yes    | No     | Yes    | No     |
| OTB               | No     | No     | Yes    | No     |
| PERIOD            | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS    | Yes    | No     | No     | No     |
| WH                | Yes    | No     | No     | No     |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| SUPS              | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**rplatupd (Replenishment Attribute Update)****Functional Area**

Replenishment

**Module Affected**

RPLATUPD.PC

**Design Overview**

The batch module reads replenishment attributes from the REPL\_ATTR\_UPDATE\_ITEM and REPL\_ATTR\_UPDATE\_LOC tables and process the item location relationships to determine what replenishment attributes for what locations have to be updated.

Replenishment attributes for each item/location are recorded in REPL\_ITEM\_LOC table. Review cycle information is kept on the REPL\_DAY table. The rejected records are written to the MC\_REJECTIONS table for later reporting.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                                 |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                                                                                                             |
| Scheduling Considerations | This program should be run before the replenishment batch programs, RPLADJ, RPLEXT, REQEXT.                                                                 |
| Pre-Processing            | Prepost rplatupd pre – truncate records in the MC_REJECTIONS table.                                                                                         |
| Post-Processing           | Prepost rplatupd post –lock and delete records from REPL_ATTR_UPDATE_ITEM, REPL_ATTR_UPDATE_LOC, REPL_ATTR_UPDATE_EXCLUDE and REPL_ATTR_UPDATE_HEAD tables. |
| Threading Scheme          | This program is threaded by location (store and warehouse).                                                                                                 |

**Restart/Recovery**

The logical unit of work is replenishment attribute id, item and location. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                    | Select | Insert | Update | Delete |
|--------------------------|--------|--------|--------|--------|
| REPL_ATTR_UPDATE_ITEM    | Yes    | No     | No     | No     |
| REPL_ATTR_UPDATE_HEAD    | Yes    | No     | No     | No     |
| REPL_ATTR_UPDATE_LOC     | Yes    | No     | No     | No     |
| ITEM_LOC                 | Yes    | No     | No     | No     |
| REPL_ITEM_LOC            | Yes    | Yes    | Yes    | Yes    |
| REPL_DAY                 | No     | Yes    | No     | Yes    |
| ITEM_SEASONS             | Yes    | Yes    | No     | No     |
| SYSTEM_OPTIONS           | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY        | Yes    | No     | No     | No     |
| ITEM_MASTER              | Yes    | No     | No     | No     |
| PACKITEM                 | Yes    | No     | No     | No     |
| DEPS                     | Yes    | No     | No     | No     |
| REPL_ITEM_LOC_UPDATES    | No     | Yes    | No     | Yes    |
| SUB_ITEMS_DETAIL         | Yes    | No     | No     | No     |
| MASTER_REPL_ATTR         | Yes    | Yes    | Yes    | Yes    |
| REPL_ATTR_UPDATE_EXCLUDE | Yes    | No     | No     | No     |
| REPL_DAY_UPDATE          | Yes    | Yes    | Yes    | No     |
| STORE_ORDERS             | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**rplbld (Replenishment Order Build)****Functional Area**

Replenishment

**Module Affected**

RPLBLD.PC

**Design Overview**

RPLBLD builds RMS orders from recommended order quantities (ROQ) generated by RPLEXT.PC and IBCALC.PC. CNTRPRSS.PC associates contracts with the ROQs created by RPLEXT.PC. These ROQs are placed on a temporary table (ORD\_TEMP or

IB\_RESULTS) by RPTEXT.PC and IBCALC.PC. All records on ORD\_TEMP/IB\_RESULTS are processed by RPLBLD each night. These ORD\_TEMP/IB\_RESULTS records are placed into logical groups, and a RMS order is created for each logical group.

In order to be placed in the same order group, the item/location ROQs from ORD\_TEMP/IB\_RESULTS must share a common supplier, have the same order\_status ('Worksheet' or 'Approved'), and be on the same contract (or not be associated with a contract). Depending on flags on the ORD\_INV\_MGMT table, two other criteria can be used for splitting order groups. First, if the INV\_MGMT\_LVL is set to 'Dept', only items in a single department are allowed in an ordering group. Secondly, the SINGLE\_LOC\_IND can be set to 'Yes'. If this is the case, only one location is allowed per ordering group. Finally, a SKU may only exist in an ordering group with a single origin country. When an item/loc ROQ ORD\_TEMP/IB\_RESULTS record is encountered with a different origin country than the one it exists with in the current ordering group, it is placed in a different ordering group.

To assist the recalculation and order scaling processes of replenishment ROQs, the REPL\_RESULTS record, associated with the ORD\_TEMP being processed, is updated with the ORDER\_NO and ALLOC\_NO that the ORD\_TEMP record was placed with. IB\_RESULTS is also updated with the ORDER\_NO.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                            |
|---------------------------|------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                                                                        |
| Scheduling Considerations | Runs after RPTEXT.PC, CNTRPRSS.PC (if contracting is being used). Runs after VRPLBLD and IBCALC. Runs before SUPCNSTR. |
| Pre-Processing            | None.                                                                                                                  |
| Post-Processing           | None.                                                                                                                  |
| Threading Scheme          | This program is threaded by supplier.                                                                                  |

### Restart/Recovery

The logical unit of work is supplier, contract number, and order status. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| ORD_TEMP              | Yes    | No     | No     | No     |
| REPL_RESULTS          | Yes    | No     | Yes    | No     |
| WH                    | Yes    | No     | No     | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| IB_RESULTS            | Yes    | No     | Yes    | No     |
| CONTRACT_HEADER       | Yes    | No     | Yes    | No     |
| CONTRACT_DETAIL       | Yes    | No     | Yes    | No     |
| ORDSKU                | Yes    | Yes    | No     | No     |
| ORDLOC                | Yes    | Yes    | No     | No     |
| ALLOC_HEADER          | No     | Yes    | No     | No     |
| ALLOC_DETAIL          | No     | Yes    | No     | No     |
| ITEM_LOC              | Yes    | No     | No     | No     |
| ORDHEAD               | No     | Yes    | Yes    | No     |
| ORD_INV_MGMT          | Yes    | Yes    | Yes    | No     |
| ORDLC                 | No     | Yes    | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | No     | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | No     | No     | Yes    | No     |
| BUYER_WKSHT_MANUAL    | No     | No     | Yes    | No     |

**I/O Specification**

N/A

**rplext (Vendor Replenishment Extraction)****Functional Area**

Replenishment

**Module Affected**

RPLEXT.PC



## Design Overview

RPLEXT (Vendor Replenishment Extraction) is the driving program for the replenishment process. It cycles through every item-location combination that is ready to be reviewed on the current day, and calculates the quantity of the item that needs to be ordered to the location. The program then writes these temporary order line items to ORD\_TEMP and REPL\_RESULTS. ORD\_TEMP is later reviewed by the module CNTRPRSS.PC in its evaluation of orders against contract types A, C, D, whereas REPL\_RESULTS is processed by RPLBLD.

## Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                         |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                                                                                                                                                     |
| Scheduling Considerations | RPLATUPD, RILMAINT, RPLADJ, REQEXT and CNTRORDB need to run before RPLEXT.<br>If contracting is being used, CNTRPRSS should run after RPLEXT; otherwise, run IBCXPL, IBCALC RPLBLD. |
| Pre-Processing            | prepost rpl pre – truncate records in ORD_TEMP and ORD_MISSED tables.                                                                                                               |
| Post-Processing           | prepost rplext post – truncate records in tables RPL_DISTRO_TMP and RPL_ALLOC_IN_TMP.                                                                                               |
| Threading Scheme          | Multiple processes of this program can be run at the same time against different departments.                                                                                       |

## Restart/Recovery

The logical unit of work is item/supplier. Records are committed to the database when commit\_max\_ctr defined in the restart\_control table is reached

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| DOMAIN_CLASS    | Yes    | No     | No     | No     |
| DOMAIN_DEPT     | Yes    | No     | No     | No     |
| DOMAIN_SUBCLASS | Yes    | No     | No     | No     |

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| ITEM_SUPP_COUNTRY | Yes    | No     | No     | No     |
| PERIOD            | Yes    | No     | No     | No     |
| REPL_DAY          | Yes    | No     | No     | No     |
| REPL_ITEM_LOC     | Yes    | No     | Yes    | No     |
| STORE             | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS    | Yes    | No     | No     | No     |
| WH                | Yes    | No     | No     | No     |
| SUPS              | Yes    | No     | No     | No     |
| SUP_INV_MGMT      | Yes    | No     | No     | No     |
| ORD_TEMP          | No     | Yes    | No     | No     |
| REPL_RESULTS      | No     | Yes    | No     | No     |

**I/O Specification**

N/A

**rplprg (Replenishment Purge)****Functional Area**

Replenishment

**Module Affected**

RPLPRG.PC

**Design Overview**

The replenishment extraction programs (RPLEXT, REQEXT) write a number of records to REPL\_RESULTS. A store order upload process populates STORE\_ORDERS. The investment buy process writes records to IB\_RESULTS and the Buyer Worksheet Form populates BUYER\_WKSHT\_MANUAL. These tables hold information that is relevant to replenishment processes. Over time, records on these tables become unneeded and must be cleared out. The replenishment purge program goes through these tables and clears out those records that are older than a predetermined number of days. The purging cycles (number of days) are maintained in SYSTEM\_OPTIONS table.

**Scheduling Constraints**

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad Hoc      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |

| Schedule Information | Description |
|----------------------|-------------|
| Post-Processing      | N/A         |
| Threading Scheme     | N/A         |

### Restart/Recovery

Because this program performs only deletes, there is no need for restart/recovery or multithreading, and there is no driving cursor. However, this program still needs an entry on RESTART\_CONTROL to determine the number of records to be deleted between commits.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS     | Yes    | No     | No     | No     |
| ALL_TAB_PARTITIONS | Yes    | No     | No     | No     |
| REPL_RESULTS       | No     | No     | No     | Yes    |
| BUYER_WKSHT_MANUAL | No     | No     | No     | Yes    |
| STORE_ORDERS       | No     | No     | No     | Yes    |
| IB_RESULTS         | No     | No     | No     | Yes    |

### I/O Specification

N/A

## rplprg\_month (Replenishment Purge)

### Functional Area

Replenishment

### Module Affected

RPLPRG\_MONTH.PC

## Design Overview

The replenishment extraction programs (RPLEXT, REQEXT) write a number of records to REPL\_RESULTS. The investment buy process writes records to IB\_RESULTS and the Buyer Worksheet Form populates BUYER\_WKSHT\_MANUAL. These tables hold information that is relevant to replenishment processes. Over time, records on these tables become unneeded and should be cleared out. The monthly replenishment purge program goes through these tables and clears out those records that are older than a predetermined number of days (maintained in SYSTEM\_OPTIONS). The eways ewInvAdjustToRMS, ewReceiptToRMS need to be shutdown when RPLPRG\_MONTH.PC is run.

## Scheduling Constraints

| Schedule Information      | Description      |
|---------------------------|------------------|
| Processing Cycle          | Ad Hoc (Monthly) |
| Scheduling Considerations | N/A              |
| Pre-Processing            | N/A              |
| Post-Processing           | N/A              |
| Threading Scheme          | N/A              |

## Restart/Recovery

Because this program performs only deletes, there is no need for restart/recovery or multithreading, and there is no driving cursor. However, this program still needs an entry on RESTART\_CONTROL to determine the number of records to be deleted between commits.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS     | Yes    | No     | No     | No     |
| ALL_TAB_PARTITIONS | Yes    | No     | No     | No     |
| REPL_RESULTS       | No     | No     | No     | Yes    |
| BUYER_WKSHT_MANUAL | No     | No     | No     | Yes    |
| STORE_ORDERS       | No     | No     | No     | Yes    |
| IB_RESULTS         | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**rplsplsplit (Replenishment/Truck Splitting)****Functional Area**

Replenishment

**Module Affected**

RPLSPLIT.PC

**Design Overview**

The purpose of this program is to select all the orders for truck splitting, which are created by the replenishment programs. The orders that are eligible are sent into the truck splitting logic and the resulting orders are created.

The orders, which are eligible for splitting, are as follows:

- The order must be created today by replenishment with ord\_inv\_mgmt.ord\_approve\_ind = 'Y'.
- The order must not be already split.
- The order must be a single location order and the location must be a warehouse.
- The order must not have any allocations.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                  |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (Daily)                                                                                                                              |
| Scheduling Considerations | This program runs nightly after the replenishment-scaling program (SUPCNSTR.PC) and before the replenishment approval program (RPLAPPRV.PC). |
| Pre-Processing            | N/A                                                                                                                                          |
| Post-Processing           | N/A                                                                                                                                          |
| Threading Scheme          | Thread by supplier                                                                                                                           |

**Restart/Recovery**

The logical unit of work for this program is set at order level. Records are committed to the database when commit\_max\_ctr defined in the RESTART\_CONTROL table is reached.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| ORDHEAD               | Yes    | Yes    | Yes    | No     |
| ORDSKU                | Yes    | Yes    | No     | Yes    |
| ORDLOC                | Yes    | Yes    | No     | Yes    |
| ORD_INV_MGMT          | Yes    | Yes    | Yes    | Yes    |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| WH                    | Yes    | No     | No     | No     |
| V_RESTART_SUPPLIER    | Yes    | No     | No     | No     |
| ALLOC_HEADER          | Yes    | Yes    | No     | Yes    |
| ALLOC_DETAIL          | Yes    | Yes    | No     | Yes    |
| ALLOC_CHRG            | No     | No     | No     | Yes    |
| ORDHEAD_REV           | No     | No     | No     | Yes    |
| ORDSKU_REV            | No     | No     | No     | Yes    |
| ORDLOC_REV            | No     | No     | No     | Yes    |
| ORDLOC_WKSHT          | No     | No     | No     | Yes    |
| ORDLOC_DISCOUNT       | No     | No     | No     | Yes    |
| ORDCUST               | No     | No     | No     | Yes    |
| ORDLC                 | No     | No     | No     | Yes    |
| DEAL_COMP_PROM        | No     | No     | No     | Yes    |
| DEAL_ITEMLOC          | No     | No     | No     | Yes    |
| DEAL_THRESHOLD        | No     | No     | No     | Yes    |
| DEAL_DETAIL           | No     | No     | No     | Yes    |
| DEAL_QUEUE            | No     | No     | No     | Yes    |
| DEAL_CALC_QUEUE       | No     | No     | No     | Yes    |
| DEAL_HEAD             | No     | No     | No     | Yes    |
| REPL_RESULTS          | No     | No     | No     | Yes    |
| REV_ORDERS            | No     | No     | No     | Yes    |
| ITEM_LOC              | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | Yes    | No     | No     | No     |
| CONTRACT_DETAIL       | No     | No     | Yes    | No     |

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| CONTRACT_HEAD      | No     | No     | Yes    | No     |
| BUYER_WKSHT_MANUAL | No     | No     | Yes    | No     |
| IB_RESULTS         | No     | No     | Yes    | No     |

**I/O Specification**

N/A

**supcnstr (Supplier Constraint Scaling)****Functional Area**

Ordering

**Module Affected**

SUPCNSTR.PC

**Design Overview**

This new batch program processes all orders eligible for scaling during the nightly replenishment run. The purpose of this program is to select all of the orders created by the replenishment programs which are eligible for scaling. Once selected, the program serves as a wrapper program and sends each order number into the supplier constraint scaling library to actually perform the scaling on the order.

The orders which are eligible for scaling are as follows:

If due order processing was used, only orders with a written date of today, origin type = 0 (replenishment order), due order processing indicator = 'Y', due order indicator = 'Y' and a scale order to constraint indicator = 'Y' is processed. This encompasses all due orders created by replenishment which have constraints associated with them.

If due order processing was not used, only orders with a written date of today, origin type = 0 (replenishment order), ord\_approve\_ind = 'Y', status = 'W' orksheet, due order processing indicator = 'N', due order indicator = 'Y', and a scale order to constraint indicator = 'Y' is processed. This encompasses all approved orders created by replenishment which have constraints associated with them.

**Scheduling Constraints**

| Schedule Information      | Description                             |
|---------------------------|-----------------------------------------|
| Processing Cycle          | Phase 3 (daily)                         |
| Scheduling Considerations | Run after rplbld and before rplsplsplit |
| Pre-Processing            | N/A                                     |
| Post-Processing           | N/A                                     |
| Threading Scheme          | Threaded by supplier                    |

**Restart/Recovery**

The logic unit of work for this program is an order number.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table        | Select | Insert | Update | Delete |
|--------------|--------|--------|--------|--------|
| ORDHEAD      | Yes    | No     | Yes    | No     |
| ORD_INV_MGMT | Yes    | No     | Yes    | No     |
| PERIOD       | Yes    | No     | No     | No     |

### I/O Specification

N/A



## RETL Extraction Programs

### Overview

RMS works in conjunction with the Oracle Retail Extract Transform and Load (RETL) framework. This architecture optimizes a high performance data processing tool that allows database batch processes to take advantage of parallel processing capabilities. The RETL framework runs and parses through the valid operators composed in XML scripts.

This chapter serves as a reference to the RETL extraction RMS programs. Note that the data in this chapter is organized according to the logic of RDW (dimension data and table data), though a retailer can use the data to suit its business needs.

For information summarizes the configuration, architecture and features for RETL programs utilized in RMS/ReSA extractions, see Volume 3 – Backend Configuration and Operations of this Operations Guide.

For more information about the RETL tool, see the latest RETL Programmer's Guide.

**Note:** RMS 12.0 is **not** integrated with Oracle Retail planning and forecasting systems (for example, Oracle Retail Demand Forecasting).

### RMS Extract Data (Based on RDW Dimension Data)

The extraction modules that were designed originally for RDW dimension data do not have an “argument” column in the table below. These modules do not require a path/file\_name parameter. These modules assume output text files are located in \${MMHOME}/data and named <DM KSH module name>.txt. If retailers wish to change this default path, they need to pass in their own path/file\_name at the command line.

| Program        | Functional Area | Source Table or File         | Schema File       | Target File or Table |
|----------------|-----------------|------------------------------|-------------------|----------------------|
| cdedtlex.ksh   | Codes           | CODE_DETAIL                  | cdedtldm.schema   | cdedtldm.txt         |
| cmptrex.ksh    | Competitor      | COMPETITOR                   | cmptrdm.schema    | cmptrdm.txt          |
| cmptrlmex.ksh  | Competitor      | COMP_STORE_LINK, CODE_DETAIL | cmptrlmdm.schema  | cmptrlmdm.txt        |
| cmptilocex.ksh | Competitor      | COMP_STORE                   | cmptilocdm.schema | cmptilocdm.txt       |
| crnycdex.ksh   | Currency Code   | CURRENCIES                   | crnycddm.schema   | crnycddm.txt         |
| employex.ksh   | Employee        | SA_EMPLOYEE                  | employdm.schema   | employdm.txt         |

| Program       | Functional Area   | Source Table or File                                                                                                                                                                                                      | Schema File          | Target File or Table |
|---------------|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|----------------------|
| orgaraex.ksh  | Organization      | AREA                                                                                                                                                                                                                      | orgaradm.s<br>chema  | orgaradm.txt         |
| orgchanex.ksh | Organization      | CHANNELS, BANNER                                                                                                                                                                                                          | orgchandm.<br>schema | orgchandm.tx<br>t    |
| orgchnex.ksh  | Organization      | CHAIN, COMPHEAD                                                                                                                                                                                                           | orgchndm.<br>schema  | orgchndm.txt         |
| orgdisex.ksh  | Organization      | DISTRICT                                                                                                                                                                                                                  | orgdisdm.s<br>chema  | orgdisdm.txt         |
| orgllmex.ksh  | Organization      | LOC_LIST_DETAIL                                                                                                                                                                                                           | orgllmdm.s<br>chema  | orgllmdm.txt         |
| orglocex.ksh  | Organization      | STORE, DISTRICT,<br>CURRENCIES,<br>COUNTRY,<br>STORE_ATTRIBUTES,<br>STORE_FORMAT,STA<br>TE, TSFZONE,<br>PROMOZONE, WH,<br>SYSTEM_OPTIONS,<br>WH_ATTRIBUTES,<br>PROMO_ZONE,<br>CHANNELS,<br>BANNER<br>USER_TAB_<br>COLUMNS | orglocdm.s<br>chema  | orglocdm.txt         |
| orglolex.ksh  | Organization      | LOC_LIST_HEAD                                                                                                                                                                                                             | orgloldm.s<br>chema  | orgloldm.txt         |
| orgltmex.ksh  | Organization      | LOC_TRAITS_<br>MATRIX                                                                                                                                                                                                     | orgltmdm.s<br>chema  | orgltmdm.txt         |
| orgltrex.ksh  | Organization      | LOC_TRAITS                                                                                                                                                                                                                | orgltrdm.sc<br>hema  | orgltrdm.txt         |
| orgrgnex.ksh  | Organization      | REGION                                                                                                                                                                                                                    | orgrgndm.s<br>chema  | orgrgndm.txt         |
| phasex.ksh    | Product<br>Season | PHASES                                                                                                                                                                                                                    | phasdm.sc<br>hema    | phasdm.txt           |
| prdclex.ksh   | Product           | CLASS, MERCHANT,<br>BUYER,DEPS                                                                                                                                                                                            | prdcldm.s<br>chema   | prdcldm.txt          |
| prdcmpex.ksh  | Company           | COMPHEAD                                                                                                                                                                                                                  | prdcmpdm.<br>schema  | prdcmpdm.tx<br>t     |

| Program         | Functional Area        | Source Table or File                                                                                  | Schema File        | Target File or Table |
|-----------------|------------------------|-------------------------------------------------------------------------------------------------------|--------------------|----------------------|
| prddepx.ksh     | Product                | DEPS, CODE_DETAIL, MERCHANT, BUYER                                                                    | prddepdm.schema    | prddepdm.txt         |
| prddiffex.ksh   | Product                | DIFF_IDS                                                                                              | prddiffdm.schema   | prddiffdm.txt        |
| prddivex.ksh    | Product                | DIVISION, COMPHEAD, MERCHANT, BUYER                                                                   | prddivdm.schema    | prddivdm.txt         |
| prddtypex.ksh   | Product                | DIFF_TYPES                                                                                            | prddtypdm.schema   | prddtypdm.txt        |
| prdgrpex.ksh    | Product                | GROUPS, MERCHANT, BUYER                                                                               | prdgrpdm.schema    | prdgrpdm.txt         |
| Prdislex.ksh    | Item-Supplier-Location | ITEM_SUPP_COUNTRY_LOC, ITEM_MASTER, ITEM_SUPP_COUNTRY_DIM, ITEM_SUPP_COUNTRY, ITEM_LOC, ITEM_SUPPLIER | prdisldm.schema    | prdisldm.txt         |
| prditmex.ksh    | Product Dimension      | ITEM_MASTER, UOM_CLASS, CODE_DETAIL, USER_TAB_COLUMNS                                                 | prditmdm.schema    | prditmdm.txt         |
| prditmlex.ksh   | Product                | SKULIST_HEAD                                                                                          | prditmldm.schema   | prditmldm.txt        |
| prditmlmex.ksh  | Product                | SKULIST_DETAIL, ITEM_MASTER                                                                           | prditmlmdm.schema  | prditmlmdm.txt       |
| prditmltmex.ksh | Item-Location Trait    | ITEM_LOC_TRAITS, ITEM_MASTER, CODE_DETAIL                                                             | prditmltmdm.schema | prditmltmdm.txt      |

| Program                                                                                                                                                                                                                                                                                                                            | Functional Area | Source Table or File                                                   | Schema File                                             | Target File or Table                           |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|------------------------------------------------------------------------|---------------------------------------------------------|------------------------------------------------|
| prditmsmex.ksh<br><b>Note:</b><br>prditmsmex.ksh extracts the latest season/phase combination for all tracking level and above items. In other words, if item A is attached to season A/phase A and season A/phase B in RMS, and phase B starts after phase A, only season A/phase B shows up in the matrix association to item A. | Product         | ITEM_SEASONS, PHASES, ITEM_MASTER                                      | prditmsmdm.schema                                       | prditmsmdm.txt                                 |
| prdpimex.ksh                                                                                                                                                                                                                                                                                                                       | Product         | PACKITEM_BREAKOUT, ITEM_MASTER                                         | prdpimdm.schema                                         | prdpimdm.txt                                   |
| prdsbcex.ksh                                                                                                                                                                                                                                                                                                                       | Product         | SUBCLASS, DEPS, CLASS, BUYER, MERCHANT                                 | prdsbcdm.schema                                         | prdsbcdm.txt                                   |
| prdudaex.ksh                                                                                                                                                                                                                                                                                                                       | Product         | ITEM_MASTER, UDA, UDA_ITEM_DATE, UDA_ITEM_FF, UDA_VALUES, UDA_ITEM_LOV | prditmuhdm.schema, prditmuddm.schema, prditmumdm.schema | prditmuhdm.txt, prditmuddm.txt, prditmumdm.txt |
| regngrpex.ksh                                                                                                                                                                                                                                                                                                                      | Regionality     | SEC_GROUP, CODE_DETAIL                                                 | regngrpdm.schema                                        | regngrpdm.txt                                  |
| regnmtxex.ksh                                                                                                                                                                                                                                                                                                                      | Regionality     | REGIONALITY_MATRIX, ITEM_MASTER, ITEM_SUPP_COUNTRY_LOC                 | regnmtxdm.schema                                        | regnmtxdm.txt                                  |

| Program          | Functional Area           | Source Table or File                                                                                           | Schema File             | Target File or Table |
|------------------|---------------------------|----------------------------------------------------------------------------------------------------------------|-------------------------|----------------------|
| Rsnex.ksh        | Reason                    | CODE_DETAIL,<br>INV_ADJ_REASON,<br>INV_STATUS_TYPES,<br>QC_FAILURE_CODES,<br>CODE_HEAD,<br>NON_MERCH_CODE_HEAD | rsndm.sche<br>ma        | rsndm.txt            |
| seasnex.ksh      | Product Season            | SEASONS                                                                                                        | seasndm.sc<br>hema      | seasndm.txt          |
| subtrantypex.ksh | Sub-Transaction Type      | CODE_DETAIL                                                                                                    | subtrantyp<br>dm.schema | subtrantypdm<br>.txt |
| supctrex.ksh     | Supplier                  | CONTRACT_<br>HEADER,<br>CODE_DETAIL                                                                            | supctrdm.s<br>chema     | supctrdm.txt         |
| Supsupex.ksh     | Supplier                  | SUPS, CURRENCIES,<br>SYSTEM_OPTIONS,<br>USER_TAB_<br>COLUMNS                                                   | supsupdm.<br>schema     | supsupdm.txt         |
| Suptrmex.ksh     | Supplier                  | SUP_TRAITS_MATRI<br>X                                                                                          | suptrmdm.<br>schema     | suptrmdm.txt         |
| suptrtex.ksh     | Supplier                  | SUP_TRAITS                                                                                                     | suptrtdm.sc<br>hema     | suptrtdm.txt         |
| tndrtypex.ksh    | Tender Type               | POS_TENDER_TYPE_<br>HEAD,<br>CODE_DETAIL                                                                       | tndrtypedm<br>.schema   | tndrtypedm.t<br>xt   |
| ttltypex.ksh     | ReSA Total Type Dimension | SA_TOTAL_HEAD                                                                                                  | ttltypdm.sc<br>hema     | ttltypdm.txt         |

## RMS Extract Data (Based on RDW Fact Data)

The 'Arguments' column lists all the command line parameters that exist in addition to the module name itself. For the extraction modules below, the data file path/file\_name is a required command line parameter. The "Arguments" column contains the extraction data file directory path and file name, such as \${MMHOME}/data/cmptrcilddm.txt. If retailers wish to change this path, they need to substitute their own path/file\_name at the command line.

Unless otherwise noted, the number of days that a transaction can be back posted is limited by the stock ledger. If a transaction is extracted with a date prior to or equal to the last closed end-of-month, then the date is changed during the extraction to the current business virtual date.

| Program           | Functional Area    | External Data Source | Source Table or File                                                   | Schema File          | Target File or Table | Arguments                 | Notes                                                                                                                                                                                                                               |
|-------------------|--------------------|----------------------|------------------------------------------------------------------------|----------------------|----------------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| cmptrprcildex.ksh | Competitor Pricing | RMS                  | COMP_STORE_LINK,COMP_PRICE_HIST,CURRENTCY_RATES,COMP_STORE,ITEM_MASTER | cmptrprcilddm.schema | cmptrprcilddm.txt    | output_file_path/filename | Back posting of competitor pricing data is not limited by the RMS stock ledger. This module accurately back posts competitor pricing facts, regardless of whether the facts occurred before the RMS SYSTEM_VARIABLES.LAST_EOM_DATE. |
| cstislindex.ksh   | Cost               | RMS                  | PRICE_HIST, ITEM_SUPP_COUNTRY_LOC, ITEM_LOC, ITEM_MASTER               | cstislddm.schema     | cstislddm.txt        | output_file_path/filename |                                                                                                                                                                                                                                     |
| exchngrate.x.ksh  | Exchange Rates     | RMS                  | CURRENCY_RATES, EURO_EXCHANGE_RATE                                     | exchngratedm.schema  | exchngratedm.txt     | output_file_path/filename |                                                                                                                                                                                                                                     |

| Program        | Functional Area      | External Data Source | Source Table or File                                                                                                   | Schema File       | Target File or Table | Arguments                 | Notes                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----------------|----------------------|----------------------|------------------------------------------------------------------------------------------------------------------------|-------------------|----------------------|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| invildex.ksh   | Inventory Position   | RMS                  | ORDLOC_REV, V_PACKSKU_QTY, IF_TRAN_DATA, ITEM_MASTER, ITEM_LOC, ITEM_LOC_SOH, REPL_ITEM_LOC, ORDHEAD, ORDLOC, PACKITEM | invilddm.schema   | invilddm.txt         | out_file_path/filename    | This module only pulls the inventory position for item-location combinations that underwent a change in inventory position for the given day. The fact that something changed is communicated to a target system (such as RDW) by a record for the item-location on the IF_TRAN_DATA table or a record on the ORDLOC_REV table for the item-location with the rev_date equal to the business virtual date for the batch run. |
| ivaildex.ksh   | Inventory Adjustment | RMS                  | IF_TRAN_DATA, ITEM_MASTER                                                                                              | ivailddm.schema   | ivailddm.txt         | out_file_path/filename    |                                                                                                                                                                                                                                                                                                                                                                                                                              |
| ivrcpildex.ksh | Inventory Receipts   | RMS                  | IF_TRAN_DATA, ITEM_MASTER                                                                                              | ivrcpilddm.schema | ivrcpilddm.txt       | out_file_path/filename    |                                                                                                                                                                                                                                                                                                                                                                                                                              |
| ivrildex.ksh   | Return to Vendor     | RMS                  | RTV_HEAD, RTV_DETAIL, ITEM_LOC, ITEM_MASTER                                                                            | ivrilddm.schema   | ivrilddm.txt         | output_file_path/filename |                                                                                                                                                                                                                                                                                                                                                                                                                              |
| ivtildex.ksh   | Inventory Transfers  | RMS                  | IF_TRAN_DATA, ITEM_MASTER                                                                                              | ivtilddm.schema   | ivtilddm.txt         | output_file_path/filename |                                                                                                                                                                                                                                                                                                                                                                                                                              |

| Program        | Functional Area                                | External Data Source | Source Table or File                                                                            | Schema File                                                                                                                        | Target File or Table | Arguments                                                     | Notes                                                                                                                                                          |
|----------------|------------------------------------------------|----------------------|-------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------|----------------------|---------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ivuildex.ksh   | Unavailable Inventory                          | RMS                  | INV_STATUS_QTY,<br>ITEM_LOC,<br>ITEM_LOC_SOH,<br>IF_TRAN_DATA,V_<br>PACKSKU_QTY,<br>ITEM_MASTER | ivuilddm.schema                                                                                                                    | ivuilddm.txt         | out_file_path/<br>filename                                    |                                                                                                                                                                |
| lptotcldex.ksh | Loss Prevention Totals (cashier over or short) | ReSA(RDWC file)      | RDWC file                                                                                       | Input (formats input data from ReSA):<br>lptotcldex.schema<br>_____Output<br>tput (formats output text file):<br>lptotclddm.schema | lptotclddm.txt       | output_file_path/<br>filename<br>input_file_path/<br>filename | 1. Input file name must begin with RDWC.<br><br>2. Before running lptotcldex, RDWC input file must be properly formatted by running ReSA Perl script resa2rdw. |
| lptotldex.ksh  | Loss Prevention Totals (user defined totals)   | ReSA(RDWS file)      | RDWS file                                                                                       | Input (formats input data from ReSA):<br>lptotldex.schema<br>_____Output<br>tput (formats output text file):<br>lptotlddm.schema   | lptotlddm.txt        | output_file_path/<br>filename<br>input_file_path/<br>filename | 1. Input file name must begin with RDWS.<br><br>2. Before running lptotldex, RDWS input file must be properly formatted by running ReSA Perl script resa2rdw.  |
| ncstuildex.ksh | Net Cost                                       | RMS                  | FUTURE_COST,<br>ITEM_SUPP_COUNTRY,<br>ITEM_LOC,<br>ITEM_MASTER                                  | ncstuilddm.schema                                                                                                                  | ncstuilddm.txt       | output_file_path/<br>filename                                 |                                                                                                                                                                |



| Program       | Functional Area       | External Data Source | Source Table or File                                                             | Schema File      | Target File or Table | Arguments                 | Notes |
|---------------|-----------------------|----------------------|----------------------------------------------------------------------------------|------------------|----------------------|---------------------------|-------|
| prcildex.ksh  | Pricing               | RMS                  | PRICE_HIST, ITEM_MASTER                                                          | prcilddm.schema  | prcilddm.txt         | output_file_path/filename |       |
| rplcildex.ksh | Replacements          | RMS                  | IF_TRAN_DATA, STORE, CHANNELS                                                    | rplcilddm.schema | rplcilddm.txt        | output_file_path/filename |       |
| savidex.ksh   | Supplier Availability | RMS                  | SUP_AVAIL                                                                        | saviddm.schema   | saviddm.txt          | output_file_path/filename |       |
| scmialdex.ksh | Supplier Compliance   | RMS                  | SHIPMENT, ORDHEAD                                                                | scmialddm.schema | scmialddm.txt        | output_file_path/filename |       |
| scmioldex.ksh | Supplier Compliance   | RMS                  | ORDHEAD, ORDLOC, STORE, WH, SHIPMENT                                             | scmiolddm.schema | scmiolddm.txt        | output_file_path/filename |       |
| scrqtdex.ksh  | Supplier Compliance   | RMS                  | SHIPMENT, ORDLOC, SHIPSKU, ORDHEAD, IF_TRAN_DATA, ITEM_MASTER, V_PACKSKU_QTY     | scrqtdm.schema   | scrqtdm.txt          | output_file_path/filename |       |
| scrtldex.ksh  | Supplier Compliance   | RMS                  | IF_TRAN_DATA, ORDHEAD, SHIPMENT, WH, SOURCE_DLVRY_SCHED, SOURCE_DLVRY_SCHED_DAYS | scrtlddm.schema  | scrtlddm.txt         | output_file_path/filename |       |

| Program      | Functional Area   | External Data Source | Source Table or File                                                                         | Schema File     | Target File or Table | Arguments                 | Notes                                                                                                                                                                                           |
|--------------|-------------------|----------------------|----------------------------------------------------------------------------------------------|-----------------|----------------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sctidex.ksh  | Supplier Contract | RMS                  | CONTRACT_HEADER,<br>CONTRACT_DETAIL,<br>CONTRACT_COST,<br>ORDHEAD,<br>ORDLOC,<br>ITEM_MASTER | sctiddm.schema  | sctiddm.txt          | output_file_path/filename | Only DWI module that can extract facts above tracking level (RMS allows contract facts at the tracking level and above, even in the same item family). Item_key can be tracking level or above. |
| sfcilwex.ksh | Sales Forecasts   | RMS                  | ITEM_FORECAST,<br>DOMAIN_DEPT,<br>ITEM_MASTER,<br>DOMAIN_CLASS,<br>DOMAIN_SUBCLASS           | sfcilwdm.schema | sfcilwdm.txt         | output_file_path/filename | This module runs weekly.                                                                                                                                                                        |

| Program       | Functional Area                | External Data Source | Source Table or File                               | Schema File                                                                                                                                                 | Target File or Table                           | Arguments                                                                                                                                                   | Notes                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|--------------------------------|----------------------|----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| slsildmex.ksh | Sales and Returns Transactions | ReSA(RDWT file)      | ITEM_MASTER, VAT_ITEM, STORE, ITEM_LOC_SO H, CLASS | Input (formats input data from ReSA): slsildmex.schema<br>_____.Outputs (formats output text files): slsildmmdm.schema lptldmmdm.schema slsprmilnmdm.schema | lptldmmdm.txt, slsildmmdm.txt slsprmilnmdm.txt | st_sls_out_file_path/st_sls_out_file<br>st_lp_out_file_path/st_lp_out_file<br>st_prm_out_file_path/st_prm_out_file<br>in_file_path/in_file<br>thread_number | <p>1. This module takes one input file from ReSA (RDWT file), and outputs three flat files: a sales transaction file, a sales transaction loss prevention file, and a sales promotion detail file.</p> <p>2. Input file name must begin with RDWT.</p> <p>3. Before running slsildmex, the RDWT input file must be properly formatted by running ReSA Perl script resa2rdw.</p> <p>4. Thread_number should uniquely identify an instance of slsildmex. This allows two different stores' RDWT files to be processed by two concurrent instances of the module.</p> |

| Program            | Functional Area                     | External Data Source | Source Table or File      | Schema File                                                                                                    | Target File or Table | Arguments                                             | Notes                                                                                                                                                           |
|--------------------|-------------------------------------|----------------------|---------------------------|----------------------------------------------------------------------------------------------------------------|----------------------|-------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Slsmkdnildex.ksh   | Markdowns                           | RMS                  | IF_TRAN_DATA, ITEM_MASTER | slsmkdnildm.schema                                                                                             | slsmkdnildm.txt      | output_file_path/filename with path                   |                                                                                                                                                                 |
| stblmthex.ksh      | Stock Ledger                        | RMS                  | MONTH_DATA                | stblmthdm.schema                                                                                               | stblmthdm.txt        | output_file_path/filename                             | This module runs weekly                                                                                                                                         |
| stblwex.ksh        | Stock Ledger                        | RMS                  | WEEK_DATA                 | stblwdm.schema                                                                                                 | stblwdm.txt          | output_file_path/filename                             | This module runs weekly.                                                                                                                                        |
| ttldmex.ksh        | Tender Transaction(Loss Prevention) | ReSA (RDWF file)     | RDWF file                 | Input (formats input data from ReSA): ttldmex.schema<br>_____Output (formats output text file): ttldmdm.schema | ttldmdm.txt          | output_file_path/filename<br>input_file_path/filename | 1. Input file name must begin with RDWF.<br><br>2. Before running ttldmex, the RDWF input file must be properly formatted by running ReSA Perl script resa2rdw. |
| vchreschdex.ksh    | Escheated Vouchers                  | RMS                  | SA_VOUCHER                | vchreschddm.schema                                                                                             | vchreschddm.txt      | output_file_path/filename                             |                                                                                                                                                                 |
| vchrmoveldsgex.ksh | Voucher Movement                    | RMS                  | SA_VOUCHER                | vchrmoveldsg.schema                                                                                            | vchrmoveldsg.txt     | output_file_path/filename                             |                                                                                                                                                                 |
| vchroutlwx.ksh     | Outstanding Vouchers                | RMS                  | SA_VOUCHER                | vchroutlwdm.schema                                                                                             | vchroutlwdm.txt      | output_file_path/filename                             | This module runs weekly.                                                                                                                                        |

## Maintenance Programs

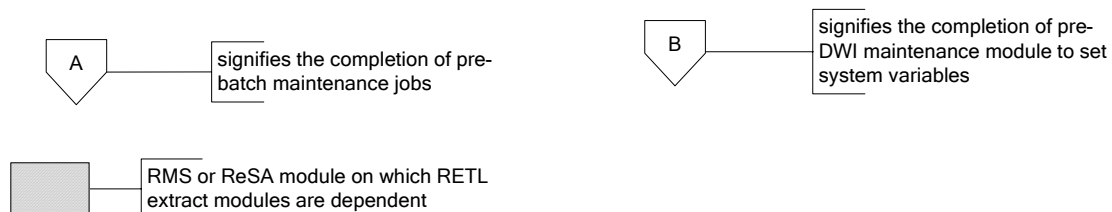
| Program             | Functional Area      | Module Type | External Data Source | Source Table or File                                         | Target File or Table                                                                                                                                                                                                                         | Notes                                                                                                                                                                                    |
|---------------------|----------------------|-------------|----------------------|--------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Post_dwi_temp.ksh   | Post-DWI maintenance | Maintenance | N/A                  | N/A                                                          | N/A                                                                                                                                                                                                                                          | Drop temp tables                                                                                                                                                                         |
| pre_dwi_extract.ksh | Pre-DWI maintenance  | Maintenance | RMS                  | PERIOD, SYSTEM_OPTIONS, SYSTEM_VARIABLES, CURRENT_CURRENCIES | class_level_vat_ind.txt, consolidation_code.txt, domain_level.txt, last_eom_date.txt, max_backpost_days.txt, multi_currency_ind.txt, prime_currency_code.txt, prime_exchange_rate.txt, stklmgr_vat_incl_retl_ind.txt, vat_ind.txt, vdate.txt | This module expects these text files to exist in \${MMHOME}/rfx/ etc when it runs. Text files containing default values for the very first run are included in the installation process. |

| Program              | Functional Area        | Module Type     | External Data Source | Source Table or File                                                                                                                                   | Target File or Table                                                                                                                            | Notes |
|----------------------|------------------------|-----------------|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|-------|
| pre_dwi_t<br>emp.ksh | Pre-DWI<br>maintenance | Maintenan<br>ce | RMS                  | CURRE<br>NCY_RA<br>TES,<br>WH,<br>EURO_E<br>XCHAN<br>GE_RAT<br>E,<br>STORE,<br>SUPS,<br>CONTR<br>ACT_HE<br>ADER,<br>INVC_D<br>ETAILE,<br>INVC_H<br>EAD | curr_tran_day_<br>temp,<br>loc_exchng_rat<br>e_temp,<br>supp_exchng_r<br>ate_temp,<br>cntrct_exchng_<br>rate_temp,<br>invc_exchng_r<br>ate_temp |       |

## Flow Diagrams and Legend

This section presents flow diagrams for data processing. The source system's program or output file is illustrated along with the program or process that interfaces with the source.

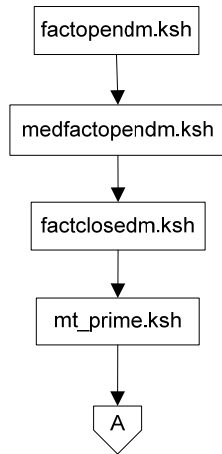
Before setting up a program schedule, familiarize yourself with the functional and technical constraints associated with each program.



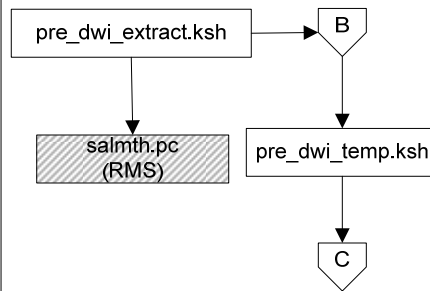
## RDW maintenance

**Note:**

The modules in this flow are RDW RETL scripts. If the retailer uses RDW, this flow must be completed before starting the pre-batch maintenance flow.

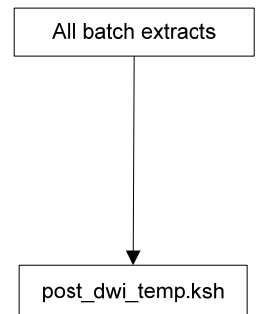


## Pre-batch maintenance

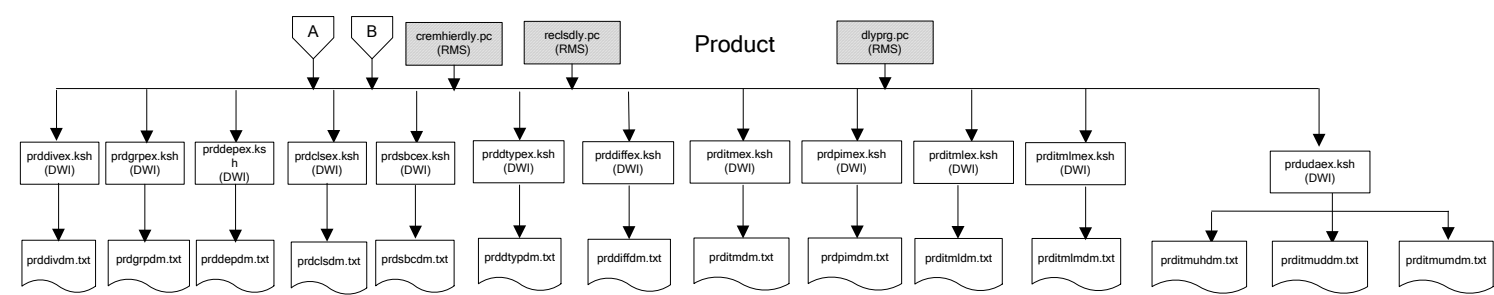
**Note:**

salmth.pc resets the last eom\_date. Thus, it must be run after the system indicator is extracted by pre\_dwi\_extract.ksh.

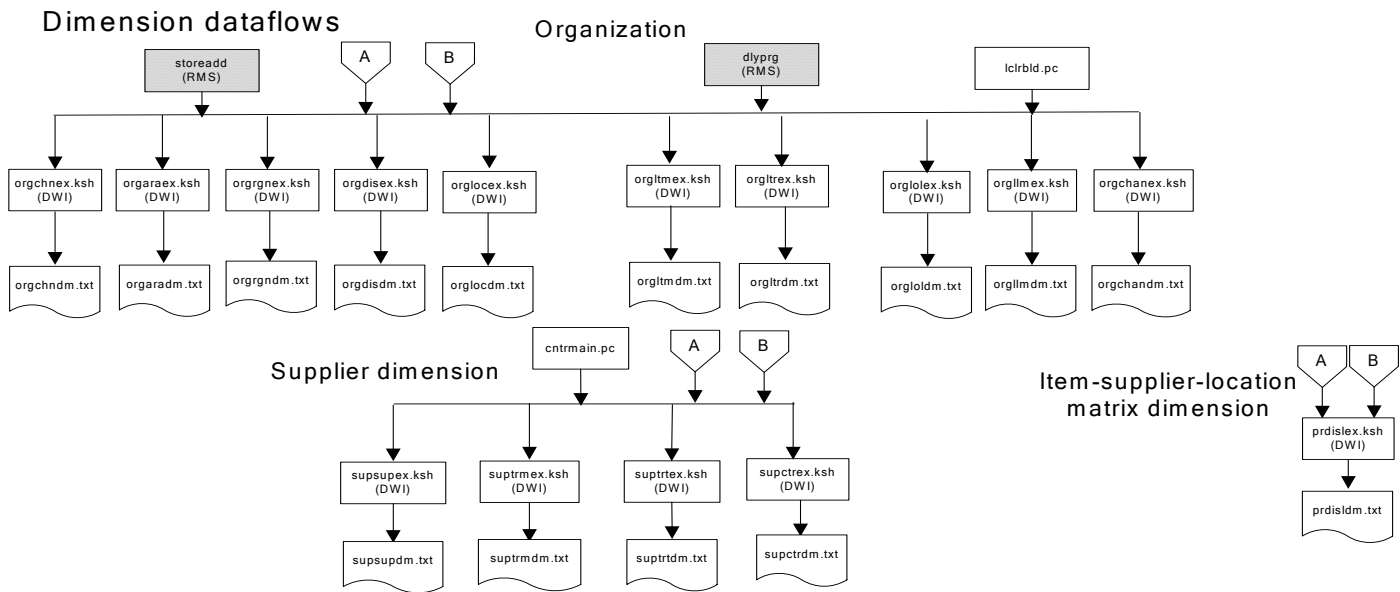
## Post-batch maintenance



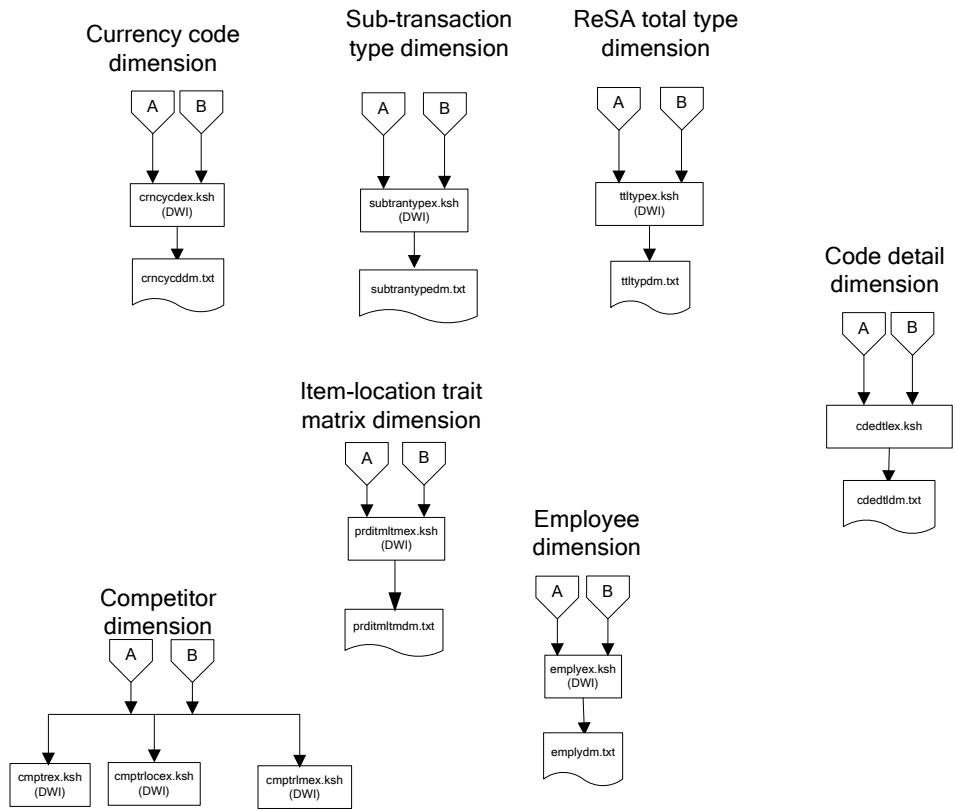
Dimension dataflows





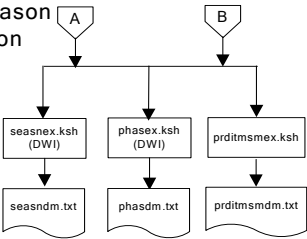


Dimension dataflows

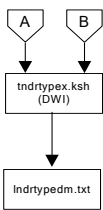


Dimension dataflows

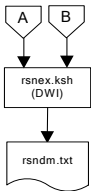
Product season dimension



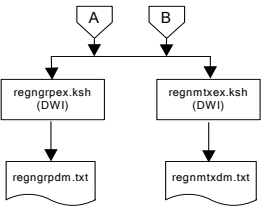
Tender type dimension

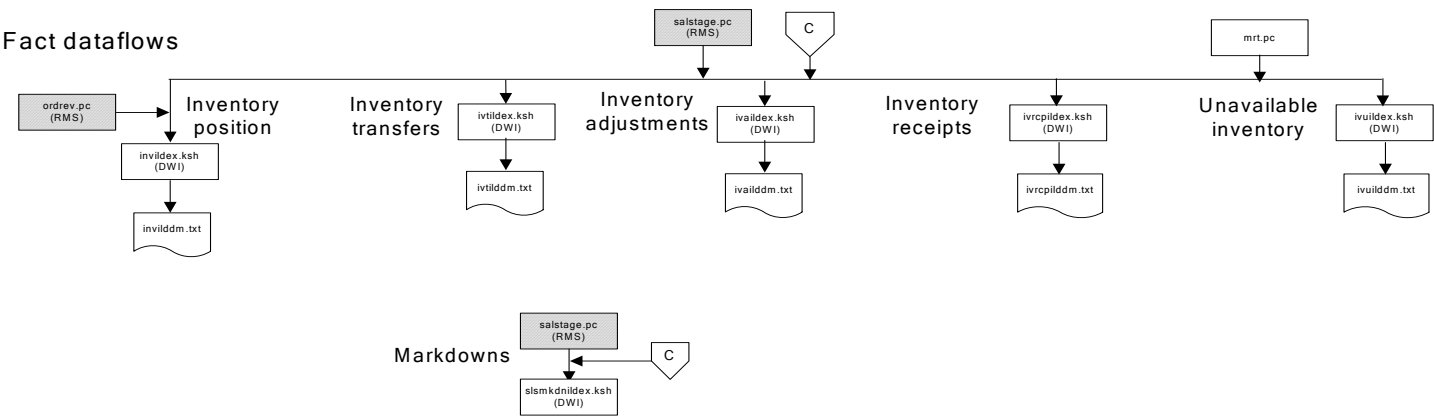


Reason dimension

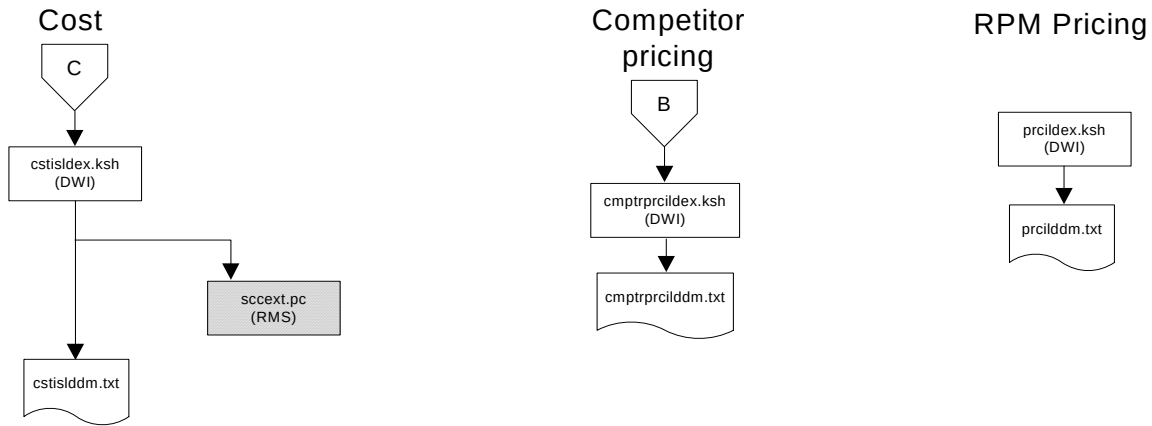


Regionality dimension

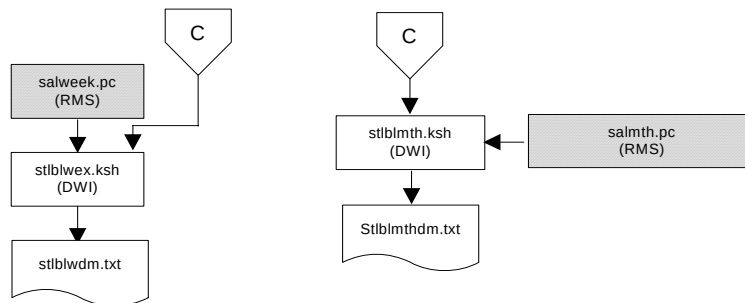




## Fact dataflows

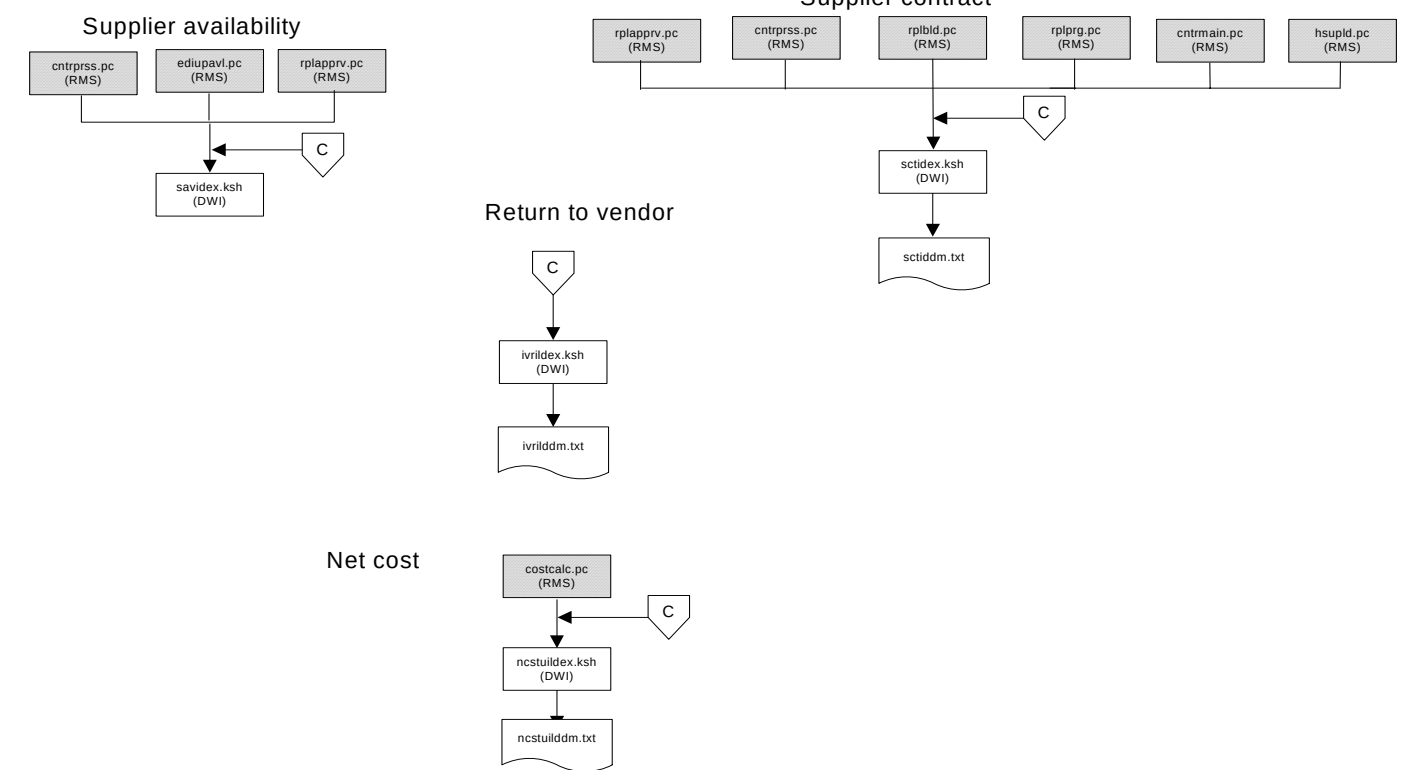


## Stock ledger



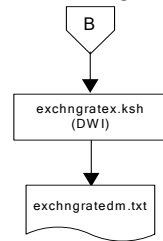
**Note:**  
Run stock ledger  
fact loads once  
weekly.

## Fact dataflows

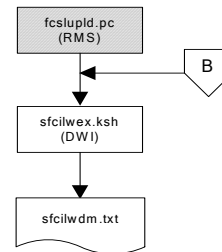


## Fact dataflows

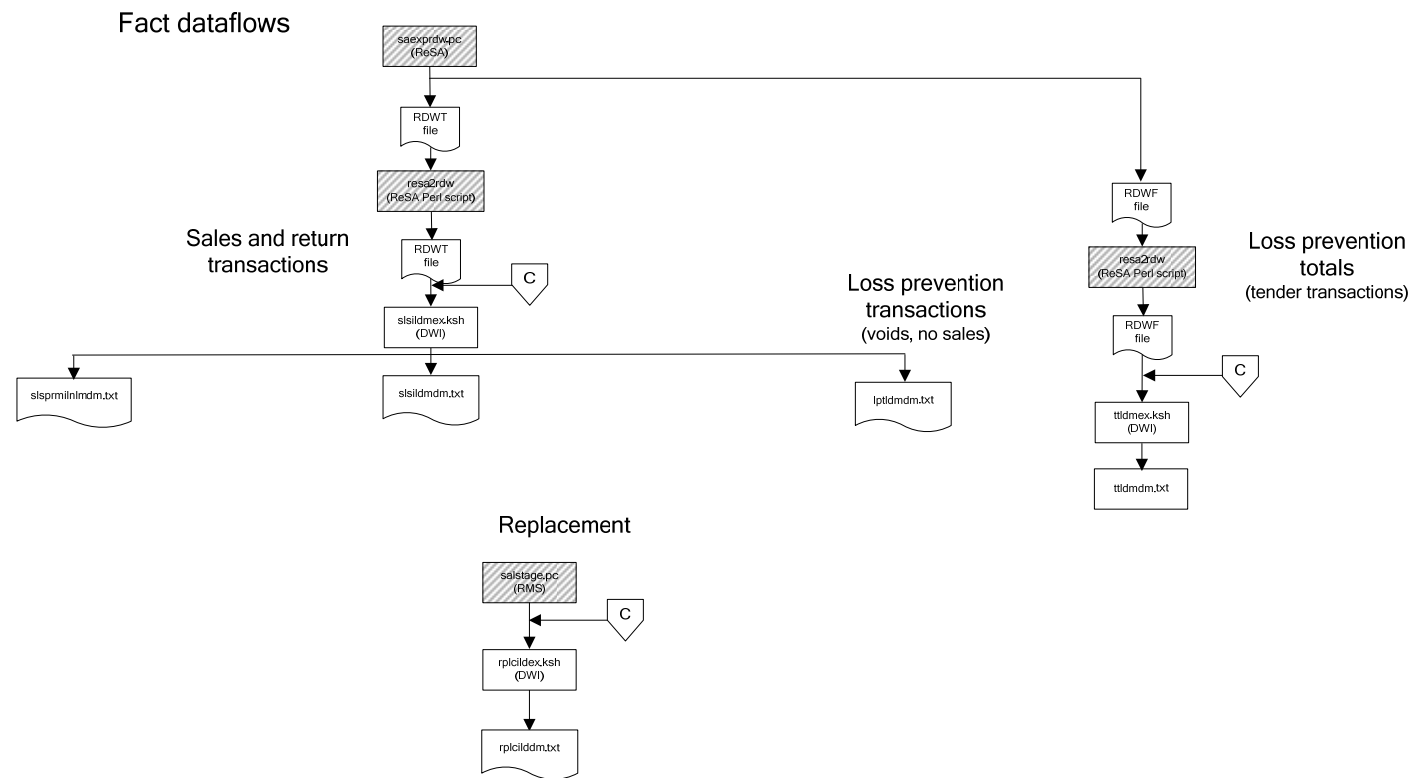
## Exchange rates



## Sales forecasts

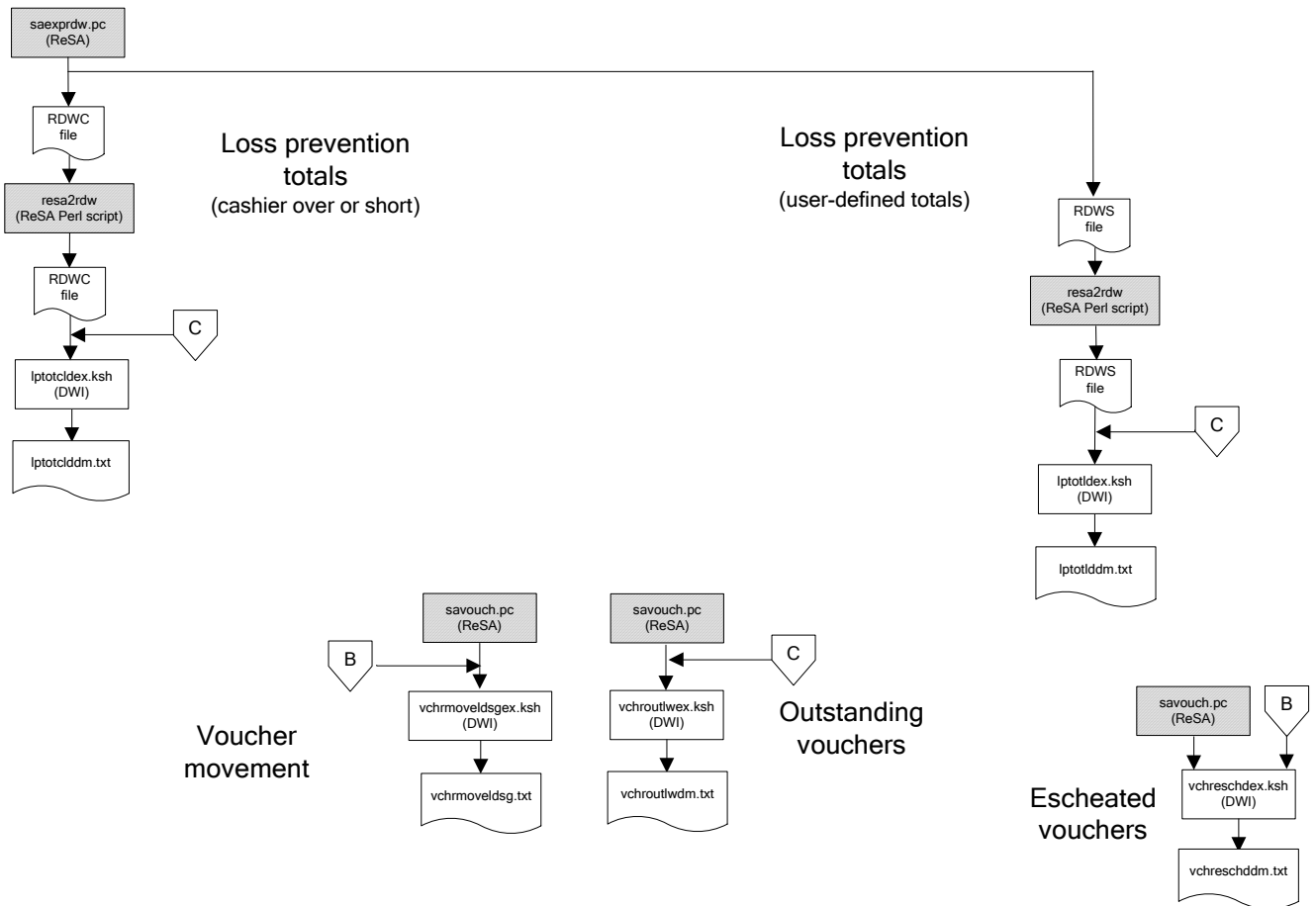


**Note:**  
Run sales forecast fact loads once weekly.

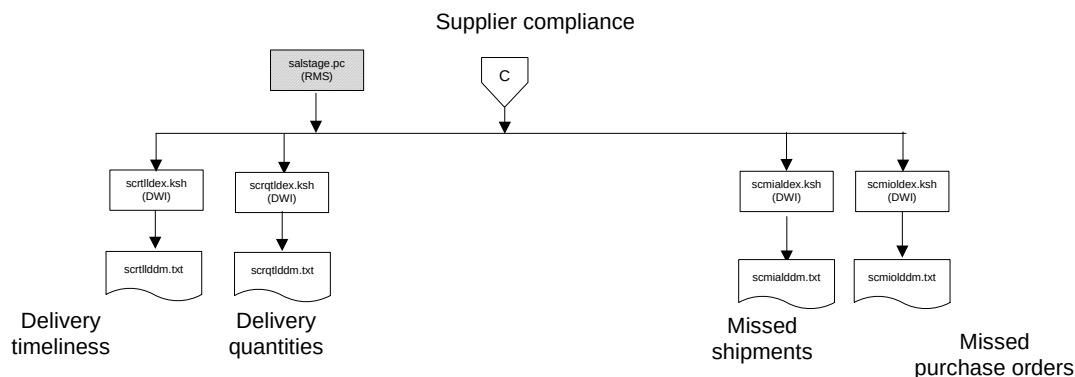




## Fact dataflows



## Fact dataflows



## RETL API Flat File Specifications

This section contains application programming interfaces (APIs) that describe the file format specifications for all text files.

In addition to providing individual field description and formatting information, the APIs provide basic business rules for the incoming data.

### API Format

Each API contains a business rules section and a file layout. Some general business rules and standards are common to all APIs. The business rules are used to ensure the integrity of the information held within RDW. In addition, each API contains a list of rules that are specific to that particular API.

### File Layout

- **Field Name:** Provides the name of the field in the text file.
- **Description:** Provides a brief explanation of the information held in the field.
- **Data Type/Bytes:** Includes both data type and maximum column length. Data type identifies one of three valid data types: character, number, or date. Bytes identifies the maximum bytes available for a field. A field may not exceed the maximum number of bytes (note that ASCII characters usually have a ratio of 1 byte = 1 character)
  - **Character:** Can hold letters (a,b,c...), numbers (1,2,3...), and special characters (\$,#,&...)
  - **Numbers:** Can hold only numbers (1,2,3...)
  - **Date:** Holds a specific year, month, day combination. The format is “YYYYMMDD”, unless otherwise specified.

- Any required formatting for a field is conveyed in the Bytes section. For example, Number (18,4) refers to number precision and scale. The first value is the precision and always matches the maximum number of digits for that field; the second value is the scale and specifies, of the total digits in the field, how many digits exist to the right of the decimal point. For example, the number –12345678901234.1234 would take up twenty ASCII characters in the flat file; however, the overall precision of the number is still (18,4).
- Field Order: Identifies the order of the field in the schema file.
- Required Field: Identifies whether the field can hold a null value. This section holds either a 'yes' or a 'no'. A 'yes' signifies the field may not hold a null value. A 'no' signifies that the field may, but is not required, to hold a null value.

## General Business Rules and Standards Common to All APIs

- Complete 'snapshot' (of what RDW refers to as dimension data):  
A majority of RDW's dimension code requires a complete view of all current dimensional data (regardless of whether the dimension information has changed) once at the end of every business day. If a complete view of the dimensional data is not provided in the text file, invalid or incorrect dimensional data can result. For instance, not including an active item in the prditmdm.txt file causes that item to be closed (as of the extract date) in the data warehouse. When a sale for the item is processed, the fact program stores not find a matching 'active' dimension record. Therefore, it is essential, unless otherwise noted in each API's specific business rules section, that a complete snapshot of the dimensional data be provided in each text file.  
If there are no records for the day, an empty flat file must still be provided.
- Updated and new records of (what RDW refers to as fact data):  
Facts being loaded to RDW can either be new or updated facts. Unlike dimension snapshots, fact flat files stores only contain new/updated facts exported from the source system once per day (or week, in some cases). Refer to each API's specific business rules section for more details.  
If there are no new or changed records for the day, an empty flat file must still be provided.
- Primary and local currency amount fields  
Amounts are stored in both primary and local currencies for most fact tables. If the source system uses multi-currency, then the primary currency column holds the primary currency amount, and the local currency column holds the local currency amount. If the location happens to use the primary currency, then both primary and local amounts hold the primary currency amount. If the source system does not use multi-currency, then only the primary currency fields are populated and the local fields hold NULL values.
- Leading/trailing values:  
Values entered into the text files are the exact values processed and loaded into the datamart tables. Therefore, the values with leading and/or trailing zeros, characters, or nulls are processed as such. RDW does not strip any of these leading or trailing values, unless otherwise noted in the individual API's business rules section.
- Indicator columns:  
Indicator columns are assumed to hold one of two values, either "Y" for yes or "N" for no.

- Delimiters:

**Note:** Make sure the delimiter is never part of your data.

- Dimension Flat File Delimiter Standards (as defined by RDW): Within dimension text files, each field must be separated by a pipe ( | ) character, for example a record from prddivdm.txt may look like the following:  
1000|1|Homewares|2006|Henry Stubbs|2302|Craig Swanson
- Fact Flat File Delimiter Standards (as defined by RDW): Within facts text files, each field must be separated by a semi-colon character ( ; ). For example a record from exchngratedm.txt may look like the following:  
WIS;20010311;1.73527820592648544918

See the latest RETL Programmer's Guide for additional information.

- End of Record Carriage Return:  
Each record in the text file must be separated by an end of line carriage return. For example, the three records below, in which each record holds four values, should be entered as:

```
1|2|3|4
5|6|7|8
9|10|11|12
```

and not as a continuous string of data, such as:

```
1|2|3|4|5|6|7|8|9|10|11|12
```

**cdedtldm.txt**

Business rules:

- This data is loaded during installation.
- This interface file contains code and code description.
- This interface file cannot contain duplicate records for a cde\_type, cde combination.
- This interface file contains the complete snapshot of active information.
- This interface file follows the dimension flat file interface layout standard.

| Name     | Description                                                                                                 | Data Type/Bytes | Field order | Required field |
|----------|-------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| CDE_TYPE | The code type, which serves as a grouping mechanism for the different codes stored on the CDE_DTL_DM table. | VARCHAR2(6)     | 1           | Yes            |
| CDE      | The unique identifier for the code within a code type.                                                      | VARCHAR2(6)     | 2           | Yes            |
| CDE_DESC | The description associated with the code.                                                                   | VARCHAR2(120)   | 3           | Yes            |

**cmptrdm.txt**

Business rules:

- This interface file contains competitor information.
- This interface file cannot contain duplicate records for a cmptr\_idnt.
- This interface file follows the dimension flat file interface layout standard.

| Name       | Description                             | Data Type/Bytes | Field order | Required field |
|------------|-----------------------------------------|-----------------|-------------|----------------|
| CMPTR_IDNT | The unique identifier of the competitor | VARCHAR2(10)    | 1           | Yes            |
| CMPTR_DESC | The name of competitor.                 | VARCHAR2(120)   | 2           | No             |
| CMPTR_ADDR | The competitor address.                 | VARCHAR2(722)   | 3           | No             |

| Name                  | Description                       | Data Type/Bytes | Field order | Required field |
|-----------------------|-----------------------------------|-----------------|-------------|----------------|
| CMPTR_CITY_NAME       | The competitor city               | VARCHAR2(120)   | 4           | No             |
| CMPTR_ST_OR_PRVNC_CDE | The competitor state or province. | VARCHAR2(3)     | 5           | No             |
| CMPTR_CNTRY_CDE       | The competitor country            | VARCHAR2(10)    | 6           | No             |

**cmptrlmdm.txt**

- This interface file defines the association between location and competitor location.
- This interface file cannot contain duplicate records for a cmptr\_loc\_idnt and cmptr\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.

| Name             | Description                                                                                                                                                                                                                | Data Type/Bytes | Field order | Required field |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_IDNT         | The unique identifier of the location.                                                                                                                                                                                     | CHARACTER(10)   | 1           | Yes            |
| CMPTR_LOC_IDNT   | The unique identifier of the competitor location.                                                                                                                                                                          | CHARACTER(10)   | 2           | Yes            |
| TARGET_CMPTR_IND | Identifies the target competitor of a retailer's store. This competitor's retail is used along with the primary store within a zone when calculating a recommended retail in Price Management. Valid values are: Y, and N. | VARCHAR2(1)     | 3           | Yes            |

| Name              | Description                                                                                           | Data Type/Bytes | Field order | Required field |
|-------------------|-------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| CMPTR_RANK        | The rank of each competitor store compared to the other stores.                                       | NUMBER(2)       | 4           | No             |
| DISTANCE          | The distance between the retailer's store and the competitor's store.                                 | NUMBER(4)       | 5           | No             |
| DISTANCE_UOM_CDE  | The unit of measure code the distance is captured in. Valid values are 1 = 'Miles', 2 = 'Kilometers'. | VARCHAR2(6)     | 6           | No             |
| DISTANCE_UOM_DESC | The unit of measure description the distance is captured in.                                          | VARCHAR2(120)   | 7           | No             |

**cmptrlocdm.txt**

Business rules:

- This interface file contains non-historical information about competitors and their individual locations.
- This interface file cannot contain duplicate records for a cmptr\_loc\_idnt, cmptr\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.

| Name           | Description                                      | Data Type/Bytes | Field order | Required field |
|----------------|--------------------------------------------------|-----------------|-------------|----------------|
| CMPTR_LOC_IDNT | The unique identifier of the competitor location | VARCHAR2(10)    | 1           | Yes            |



| Name                      | Description                                                                                   | Data Type/Bytes | Field order | Required field |
|---------------------------|-----------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| CMPTR_IDNT                | The unique identifier of the competitor                                                       | VARCHAR2(10)    | 2           | Yes            |
| CMPTR_LOC_DESC            | The competitor store description                                                              | VARCHAR2(160)   | 3           | No             |
| CMPTR_LOC_ADDR            | The competitor store's address                                                                | VARCHAR2(722)   | 4           | No             |
| CMPTR_LOC_CITY_NAME       | The competitor store city                                                                     | VARCHAR2(120)   | 5           | No             |
| CMPTR_LOC_ST_OR_PRVNC_CDE | The competitor store state                                                                    | VARCHAR2(3)     | 6           | No             |
| CMPTR_LOC_CNTRY_CDE       | The competitor store country                                                                  | VARCHAR2(10)    | 7           | No             |
| ESTIMATED_VOLUME          | The estimated yearly sales volume of the competitor at assigned location.                     | NUMBER(18,4)    | 8           | No             |
| CMPTR_CRNCY_CDE_IDNT      | The unique identifier of the currency code. E.g: USD is the local currency code for US Dollar | VARCHAR2(10)    | 9           | No             |

**cmptrprcildm.txt**

Business rules:

- This interface file contains competitor's pricing facts for the client location, competitor location and item combination on a given day.

- This interface file cannot contain duplicate transactions for item\_idnt, loc\_idnt, cmptr\_loc\_idnt, day\_dt combinations.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name                           | Description                                                                                                                                                     | Data Type/Bytes | Field order | Required field |
|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT                      | The unique identifier of an item.                                                                                                                               | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT                       | The unique identifier of the location.                                                                                                                          | CHARACTER(10)   | 2           | Yes            |
| CMPTR_LOC_IDNT                 | The unique identifier of the competitor location.                                                                                                               | CHARACTER(10)   | 3           | Yes            |
| DAY_DT                         | The calendar day on which the transaction occurred.                                                                                                             | DATE            | 4           | Yes            |
| F_CMPTR_UNIT_RTL_AMT           | The competitor's unit retail amount for a particular item in primary currency.                                                                                  | NUMBER(18,4)    | 5           | No             |
| F_CMPTR_UNIT_RTL_AMT_LCL       | The competitor's unit retail amount for a particular item in local currency.                                                                                    | NUMBER(18,4)    | 6           | No             |
| F_CMPTR_MULTI_UNIT_RTL_AMT     | The competitor's multi unit retail amount for a particular item in primary currency.                                                                            | NUMBER(18,4)    | 7           | No             |
| F_CMPTR_MULTI_UNIT_RTL_AMT_LCL | The competitor's multi unit retail amount for a particular item in local currency.                                                                              | NUMBER(18,4)    | 8           | No             |
| RTL_TYPE_CDE                   | The price type ('R'egular, 'P'romotion, 'C'learance).                                                                                                           | CHARACTER(2)    | 9           | Yes            |
| OFFER_TYPE_CDE                 | This non-aggregatable field identifies the offer type code of the competitor's promotional retail. Examples of valid values are 1 = 'Coupon', 2= 'Mailer', etc. | VARCHAR2(6)     | 10          | No             |
| MULTI_UNITS_QTY                | This non-aggregatable field identifies the multi units associated with F_CMPTR_UNIT_RTL_AMT for a particular item.                                              | NUMBER(12,4)    | 11          | No             |

**crncycddm.txt**

Business rules:

- This interface file contains currency code information.
- This interface file cannot contain duplicate records for a crncy\_cde\_idnt.
- This interface file follows the dimension flat file interface layout standard.

| Name           | Description                                                                       | Data Type/Bytes | Field order | Required field |
|----------------|-----------------------------------------------------------------------------------|-----------------|-------------|----------------|
| CRNCY_CDE_IDNT | The unique identifier of the currency code.                                       | VARCHAR2(10)    | 1           | Yes            |
| CRNCY_CDE_DESC | The description of local currency code. That is, description for USD = US Dollar. | VARCHAR2(120)   | 2           | Yes            |

**cstislddm.txt**

## Business rules:

- This interface file contains cost information for a tracking level item, supplier, and location combination on a given day.
- This interface file cannot contain duplicate transactions for an item\_idnt, loc\_idnt, supp\_idnt and day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.
- This interface file only contains records with tran\_type of 0 (new cost) or tran\_type of 2 (cost change).

| Name                | Description                                         | Data Type/Bytes | Field order | Required field |
|---------------------|-----------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT           | The unique identifier of an item.                   | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT            | The unique identifier of the location.              | CHARACTER(10)   | 2           | Yes            |
| SUPP_IDNT           | The unique identifier of a supplier.                | CHARACTER(10)   | 3           | Yes            |
| DAY_DT              | The calendar day on which the transaction occurred. | DATE            | 4           | Yes            |
| F_BASE_COST_AMT     | The cost valuation in primary currency              | NUMBER(18,4)    | 5           | No             |
| F_BASE_COST_AMT_LCL | The cost valuation in local currency                | NUMBER(18,4)    | 6           | No             |

**emptydm.txt**

Business rules:

- This interface file contains the employee data.
- This interface file cannot contain duplicate records for an empty\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name       | Description                                                                           | Data Type/Bytes | Field order | Required field |
|------------|---------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| EMPTY_IDNT | The unique identifier of the employee.                                                | VARCHAR2(10)    | 1           | Yes            |
| EMPTY_NAME | The name of the employee.                                                             | VARCHAR2(120)   | 2           | Yes            |
| EMPTY_ROLE | Indicates the type of position the employee holds. 'C'ashier, 'S'alesperson, 'O'ther. | VARCHAR2(1)     | 3           | Yes            |

**exchngratedm.txt**

Business rules:

- This interface file contains currency exchange rate information.
- This interface file cannot contain duplicate records for a crncy\_cde\_idnt, day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains only the current day's new or changed information.

| Name           | Description                                         | Data Type/Bytes | Field order | Required field |
|----------------|-----------------------------------------------------|-----------------|-------------|----------------|
| CRNCY_CDE_IDNT | The unique identifier of the currency code.         | CHARACTER(10)   | 1           | Yes            |
| DAY_DT         | The calendar day on which the transaction occurred. | DATE            | 2           | Yes            |
| F_EXCHNG_RATE  | The current exchange rate.                          | NUMBER(18,4)    | 3           | No             |

**invilddm.txt**

## Business rules:

- This interface file contains end of day inventory levels and status for an item and location combination on a given day.
- This interface file cannot contain duplicate records for an item\_idnt, loc\_idnt, day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains only the current day's new or changed information.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name         | Description                                                           | Data Type/Bytes | Field order | Required field |
|--------------|-----------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT    | The unique identifier of an item.                                     | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT     | The unique identifier of the location.                                | CHARACTER(10)   | 2           | Yes            |
| DAY_DT       | The calendar day on which the transaction occurred.                   | DATE            | 3           | Yes            |
| LOC_TYPE_CDE | The code that indicates whether the location is a store or warehouse. | CHARACTER(2)    | 4           | Yes            |
| RTL_TYPE_CDE | The price type ('R'egular, 'P'romotion, 'C'learance).                 | CHARACTER(2)    | 5           | Yes            |
| F_I_SOH_QTY  | The total quantity of inventory on hand.                              | NUMBER(12,4)    | 6           | No             |

| Name                 | Description                                                                                                                                                              | Data Type/Bytes | Field order | Required field |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_SOH_COST_AMT     | The extended cost amount of inventory in stock in primary currency. The product of the weighted average cost in primary currency and the current stock on hand quantity. | NUMBER(18,4)    | 7           | No             |
| F_I_SOH_COST_AMT_LCL | The extended cost amount of inventory in stock in local currency. The product of the weighted average cost in local currency and the current stock on hand quantity.     | NUMBER(18,4)    | 8           | No             |
| F_I_SOH_RTL_AMT      | The extended retail amount of inventory in stock in primary currency. The product of the unit retail in primary currency and the current stock on hand quantity.         | NUMBER(18,4)    | 9           | No             |

| Name                    | Description                                                                                                                                                   | Data Type/Bytes | Field order | Required field |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_SOH_RTL_AMT_LCL     | The extended retail amount of inventory in stock in local currency. The product of the unit retail in local currency and the current stock on hand quantity.  | NUMBER(18,4)    | 10          | No             |
| F_I_ON_ORD_QTY          | The quantity of inventory on order.                                                                                                                           | NUMBER(12,4)    | 11          | No             |
| F_I_ON_ORD_COST_AMT     | The extended cost amount of inventory on order in primary currency. The product of the order unit cost in primary currency and the current on order quantity. | NUMBER(18,4)    | 12          | No             |
| F_I_ON_ORD_COST_AMT_LCL | The extended cost amount of inventory on order in local currency. The product of the order unit cost in local currency and the current on order quantity.     | NUMBER(18,4)    | 13          | No             |



| Name                   | Description                                                                                                                                                             | Data Type/Bytes | Field order | Required field |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_ON_ORD_RTL_AMT     | The extended retail amount of inventory on order in primary currency. The product of the order unit retail in primary currency and the current on order quantity.       | NUMBER(18,4)    | 14          | No             |
| F_I_ON_ORD_RTL_AMT_LCL | The extended retail amount of inventory on order in local currency. The product of the order unit retail in local currency and the current on order quantity.           | NUMBER(18,4)    | 15          | No             |
| F_I_IN_TRNST_QTY       | The total quantity of inventory in transit.                                                                                                                             | NUMBER(12,4)    | 16          | No             |
| F_I_IN_TRNST_COST_AMT  | The extended cost amount of inventory in transit in primary currency. The product of the weighted average cost in primary currency and the current in transit quantity. | NUMBER(18,4)    | 17          | No             |

| Name                      | Description                                                                                                                                                         | Data Type/Bytes | Field order | Required field |
|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_IN_TRNST_COST_AMT_LCL | The extended cost amount of inventory in transit in local currency. The product of the weighted average cost in local currency and the current in transit quantity. | NUMBER(18,4)    | 18          | No             |
| F_I_IN_TRNST_RTL_AMT      | The extended retail amount of inventory in transit in primary currency. The product of the unit retail in primary currency and the current in transit quantity.     | NUMBER(18,4)    | 19          | No             |
| F_I_IN_TRNST_RTL_AMT_LCL  | The extended retail amount of inventory in transit in local currency. The product of the unit retail in local currency and the current in transit quantity.         | NUMBER(18,4)    | 20          | No             |

| Name                       | Description                                                                                                                                                                               | Data Type/Bytes | Field order | Required field |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_ALLOC_RSV_QTY          | The allocated reserved quantity. The warehouse-to-store reserved quantity, composed of reserved quantity for allocations and the reserved quantity for transfers from warehouse to store. | NUMBER(12,4)    | 21          | No             |
| F_I_ALLOC_RSV_COST_AMT     | The allocated reserved extended cost amount in primary currency. The product of the weighted average cost in primary currency and the current allocated reserved quantity.                | NUMBER(18,4)    | 22          | No             |
| F_I_ALLOC_RSV_COST_AMT_LCL | The allocated reserved extended cost amount in local currency. The product of the weighted average cost in local currency and the current allocated reserved quantity.                    | NUMBER(18,4)    | 23          | No             |

| Name                      | Description                                                                                                                                                        | Data Type/Bytes | Field order | Required field |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_ALLOC_RSV_RTL_AMT     | The allocated reserved extended retail amount in primary currency. The product of the unit retail in primary currency and the current allocated reserved quantity. | NUMBER(18,4)    | 24          | No             |
| F_I_ALLOC_RSV_RTL_AMT_LCL | The allocated reserved extended retail amount in local currency. The product of the unit retail in local currency and the current allocated reserved quantity.     | NUMBER(18,4)    | 25          | No             |
| F_I_TRNSFR_RSV_QTY        | The transfer reserved quantity. The store-to-store reserved quantity, composed of the quantity of transfers from store to store that are not shipped.              | NUMBER(12,4)    | 26          | No             |

| Name                        | Description                                                                                                                                                              | Data Type/Bytes | Field order | Required field |
|-----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_TRNSFR_RSV_COST_AMT     | The transfer reserved extended cost amount in primary currency. The product of the weighted average cost in primary currency and the current transfer reserved quantity. | NUMBER(18,4)    | 27          | No             |
| F_I_TRNSFR_RSV_COST_AMT_LCL | The transfer reserved extended cost amount in local currency. The product of the weighted average cost in local currency and the current transfer reserved quantity.     | NUMBER(18,4)    | 28          | No             |
| F_I_TRNSFR_RSV_RTL_AMT      | The transfer reserved extended retail amount in primary currency. The product of the unit retail in primary currency and the current transfer reserved quantity.         | NUMBER(18,4)    | 29          | No             |

| Name                       | Description                                                                                                                                                  | Data Type/Bytes | Field order | Required field |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_TRNSFR_RSV_RTL_AMT_LCL | The transfer reserved extended retail amount in local currency. The product of the unit retail in local currency and the current transfer reserved quantity. | NUMBER(18,4)    | 30          | No             |
| F_I_REPL_ACTV_FLAG         | Flag to indicate if end date of this record's time period is within the active and inactive dates for replenishment.                                         | VARCHAR2(1)     | 31          | No             |
| F_I_REPL_CALC_MTHD_CDE     | This column holds the replenishment method code value.                                                                                                       | VARCHAR2(2)     | 32          | No             |
| F_I_MIN_SOH_QTY            | The minimum stock on hand quantity.                                                                                                                          | NUMBER(12,4)    | 33          | No             |

| Name                     | Description                                                                                                                                                                         | Data Type/Bytes | Field order | Required field |
|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_MIN_SOH_COST_AMT     | The extended cost amount of minimum stock on hand in primary currency. The product of the average weighted cost in primary currency and the current minimum stock on hand quantity. | NUMBER(18,4)    | 34          | No             |
| F_I_MIN_SOH_COST_AMT_LCL | The extended cost amount of minimum stock on hand in local currency. The product of the average weighted cost in local currency and the current minimum stock on hand quantity.     | NUMBER(18,4)    | 35          | No             |
| F_I_MIN_SOH_RTL_AMT      | The extended retail amount of minimum stock on hand in primary currency. The product of the unit retail in primary currency and the current minimum stock on hand quantity.         | NUMBER(18,4)    | 36          | No             |

| Name                    | Description                                                                                                                                                                         | Data Type/Bytes | Field order | Required field |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_MIN_SOH_RTL_AMT_LCL | The extended retail amount of minimum stock on hand in local currency. The product of the unit retail in local currency and the current minimum stock on hand quantity.             | NUMBER(18,4)    | 37          | No             |
| F_I_MAX_SOH_QTY         | The maximum stock on hand quantity.                                                                                                                                                 | NUMBER(12,4)    | 38          | No             |
| F_I_MAX_SOH_COST_AMT    | The extended cost amount of maximum stock on hand in primary currency. The product of the average weighted cost in primary currency and the current maximum stock on hand quantity. | NUMBER(18,4)    | 39          | No             |



| Name                     | Description                                                                                                                                                                     | Data Type/Bytes | Field order | Required field |
|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_MAX_SOH_COST_AMT_LCL | The extended cost amount of maximum stock on hand in local currency. The product of the average weighted cost in local currency and the current maximum stock on hand quantity. | NUMBER(18,4)    | 40          | No             |
| F_I_MAX_SOH_RTL_AMT      | The extended retail amount of maximum stock on hand in primary currency. The product of the unit retail in primary currency and the current maximum stock on hand quantity.     | NUMBER(18,4)    | 41          | No             |
| F_I_MAX_SOH_RTL_AMT_LCL  | The extended retail amount of maximum stock on hand in local currency. The product of the unit retail in local currency and the current maximum stock on hand quantity.         | NUMBER(18,4)    | 42          | No             |

| Name                 | Description                                                                                                                                 | Data Type/Bytes | Field order | Required field |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_INCR_PCT         | The replenishment incremental percentage or multiple value.                                                                                 | NUMBER(12,4)    | 43          | No             |
| F_I_COST_AMT         | The weighted average cost for stock in primary currency.                                                                                    | NUMBER(18,4)    | 44          | No             |
| F_I_COST_AMT_LCL     | The weighted average cost for stock in local currency.                                                                                      | NUMBER(18,4)    | 45          | No             |
| F_I_STD_COST_AMT     | The cost of the latest item supplied in primary currency.<br>Used to reflect the difference in unit cost if cost method accounting is used. | NUMBER(18,4)    | 46          | No             |
| F_I_STD_COST_AMT_LCL | The cost of the latest item supplied in local currency.<br>Used to reflect the difference in unit cost if cost method accounting is used.   | NUMBER(18,4)    | 47          | No             |
| F_I_RTL_AMT          | The corporate unit purchase price for stock in primary currency.                                                                            | NUMBER(18,4)    | 48          | No             |

| Name               | Description                                                                                                                                                                     | Data Type/Bytes | Field order | Required field |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_RTL_AMT_LCL    | The corporate unit purchase price for stock in local currency.                                                                                                                  | NUMBER(18,4)    | 49          | No             |
| F_I_AGED_30_60_QTY | This column is not populated in the base version of RDW. This fact is used to record the quantity of inventory that is between 30 and 60 days old at this location on this day. | NUMBER(12,4)    | 50          | No             |
| F_I_AGED_61_90_QTY | This column is not populated in the base version of RDW. This fact is used to record the quantity of inventory that is between 61 and 90 days old at this location on this day. | NUMBER(12,4)    | 51          | No             |

| Name                  | Description                                                                                                                                                                      | Data Type/Bytes | Field order | Required field |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_AGED_91_120_QTY   | This column is not populated in the base version of RDW. This fact is used to record the quantity of inventory that is between 91 and 120 days old at this location on this day. | NUMBER(12,4)    | 52          | No             |
| F_I_AGED_121_QTY      | This column is not populated in the base version of RDW. This fact is used to record the quantity of inventory that is 121days old or older at this location on this day.        | NUMBER(12,4)    | 53          | No             |
| F_I_SLS_ADMN_COST_AMT | This fact could be used to store additional cost information for this item, location, and day relationship. Sales and admin cost.                                                | NUMBER(18,4)    | 54          | No             |

| Name              | Description                                                                                                                                                                             | Data Type/Bytes | Field order | Required field |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_DIST_COST_AMT | This column is not populated in the base version of RDW. This fact could be used to store additional cost information for this item, location, and day relationship. Supply chain cost. | NUMBER(18,4)    | 55          | No             |

**ivailddm.txt**

Business rules:

- This interface file contains the inventory adjustment data for an item, location, and reason combination on a given day.
- This interface file cannot contain duplicate transactions for an item\_idnt, loc\_idnt, reasn\_type\_idnt, reasn\_cde\_idnt, and day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name         | Description                                                           | Data Type/Bytes | Field order | Required field |
|--------------|-----------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT    | The unique identifier of an item.                                     | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT     | The unique identifier of the location.                                | CHARACTER(10)   | 2           | Yes            |
| LOC_TYPE_CDE | The code that indicates whether the location is a store or warehouse. | CHARACTER(2)    | 3           | Yes            |

| Name                 | Description                                                              | Data Type/Bytes | Field order | Required field |
|----------------------|--------------------------------------------------------------------------|-----------------|-------------|----------------|
| DAY_DT               | The calendar day on which the transaction occurred.                      | DATE            | 4           | Yes            |
| F_I_ADJ_QTY          | The quantity of the adjustment to the total stock on hand.               | NUMBER(12,4)    | 5           | No             |
| F_I_ADJ_COST_AMT     | The cost amount of total stock on hand adjustment in primary currency.   | NUMBER(18,4)    | 6           | No             |
| F_I_ADJ_COST_AMT_LCL | The cost amount of total stock on hand adjustment in local currency.     | NUMBER(18,4)    | 7           | No             |
| F_I_ADJ_RTL_AMT      | The retail amount of total stock on hand adjustment in primary currency. | NUMBER(18,4)    | 8           | No             |
| F_I_ADJ_RTL_AMT_LCL  | The retail amount of total stock on hand adjustment in local currency.   | NUMBER(18,4)    | 9           | No             |
| REASN_TYPE_IDNT      | The unique identifier of the reason type.                                | CHARACTER(6)    | 10          | Yes            |
| REASN_CODE_IDNT      | The unique identifier of the reason code.                                | CHARACTER(6)    | 11          | Yes            |

**ivrcpilddm.txt**

## Business rules:

- This interface file contains inventory receipts for an item and location combination on a given day.
- This interface file cannot contain duplicate transactions for an item\_idnt, loc\_idnt, and day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name                   | Description                                         | Data Type/Bytes | Field order | Required field |
|------------------------|-----------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT              | The unique identifier of an item.                   | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT               | The unique identifier of the location.              | CHARACTER(10)   | 2           | Yes            |
| DAY_DT                 | The calendar day on which the transaction occurred. | DATE            | 3           | Yes            |
| F_I_RCPTS_QTY          | The receipt quantity.                               | NUMBER(12,4)    | 4           | No             |
| F_I_RCPTS_COST_AMT     | The receipt cost amount in primary currency.        | NUMBER(18,4)    | 5           | No             |
| F_I_RCPTS_COST_AMT_LCL | The receipt cost amount in local currency.          | NUMBER(18,4)    | 6           | No             |
| F_I_RCPTS_RTL_AMT      | The receipt retail amount in primary currency.      | NUMBER(18,4)    | 7           | No             |
| F_I_RCPTS_RTL_AMT_LCL  | The receipt retail amount in local currency.        | NUMBER(18,4)    | 8           | No             |

**ivriddm.txt**

Business rules:

- This interface file contains data on inventory returned to a supplier for a supplier, item, reason, and location combination on a given day.
- This interface file cannot contain duplicate transactions for an item\_idnt, supp\_idnt, loc\_idnt, and day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name                 | Description                                                           | Data Type/Bytes | Field order | Required field |
|----------------------|-----------------------------------------------------------------------|-----------------|-------------|----------------|
| SUPP_IDNT            | The unique identifier of a supplier.                                  | CHARACTER(10)   | 1           | Yes            |
| ITEM_IDNT            | The unique identifier of an item.                                     | CHARACTER(25)   | 2           | Yes            |
| LOC_IDNT             | The unique identifier of the location.                                | CHARACTER(10)   | 3           | Yes            |
| LOC_TYPE_CDE         | The code that indicates whether the location is a store or warehouse. | CHARACTER(2)    | 4           | Yes            |
| DAY_DT               | The calendar day on which the transaction occurred.                   | DATE            | 5           | Yes            |
| F_I_RTV_QTY          | The quantity of the stock returned to vendor.                         | NUMBER(12,4)    | 6           | No             |
| F_I_RTV_COST_AMT     | The cost of the stock returned to vendor in primary currency.         | NUMBER(18,4)    | 7           | No             |
| F_I_RTV_COST_AMT_LCL | The cost of the stock returned to vendor in local currency.           | NUMBER(18,4)    | 8           | No             |



| Name                | Description                                                             | Data Type/Bytes | Field order | Required field |
|---------------------|-------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_RTV_RTL_AMT     | The retail amount of the stock returned to vendor, in primary currency. | NUMBER(18,4)    | 9           | No             |
| F_I_RTV_RTL_AMT_LCL | The retail amount of the stock returned to vendor, in local currency.   | NUMBER(18,4)    | 10          | No             |
| REASN_TYPE_IDNT     | The unique identifier of the reason type.                               | CHARACTER(6)    | 11          | Yes            |
| REASN_CODE_IDNT     | The unique identifier of the reason code.                               | CHARACTER(6)    | 12          | Yes            |

**ivtilddm.txt**

Business rules:

- This interface file contains inventory transfers for an item, from-location, to-location, and transfer type combination on a given day.
- This interface file cannot contain duplicate transactions for an item\_idnt, loc\_idnt, from\_loc\_idnt, tsf\_type\_cde, and day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name          | Description                                                   | Data Type/Bytes | Field order | Required field |
|---------------|---------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT     | The unique identifier of an item.                             | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT      | The unique identifier of the location.                        | CHARACTER(10)   | 2           | Yes            |
| FROM_LOC_IDNT | The unique identifier for a source location for the transfer. | CHARACTER(10)   | 3           | Yes            |
| TSF_TYPE_CDE  |                                                               | CHARACTER(2)    | 4           | Yes            |

| Name                        | Description                                                                                | Data Type/Bytes | Field order | Required field |
|-----------------------------|--------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| DAY_DT                      | The calendar day on which the transaction occurred.                                        | DATE            | 5           | Yes            |
| F_I_TSF_TO_LOC_QTY          | The quantity transferred to a destination location.                                        | NUMBER(12,4)    | 6           | No             |
| F_I_TSF_TO_LOC_COST_AMT     | The transfer cost amount for a destination location in primary currency.                   | NUMBER(18,4)    | 7           | No             |
| F_I_TSF_TO_LOC_COST_AMT_LCL | The transfer cost amount for a destination location in the destination's local currency.   | NUMBER(18,4)    | 8           | No             |
| F_I_TSF_TO_LOC_RTL_AMT      | The transfer retail amount for a destination location in primary currency.                 | NUMBER(18,4)    | 9           | No             |
| F_I_TSF_TO_LOC_RTL_AMT_LCL  | The transfer retail amount for a destination location in the destination's local currency. | NUMBER(18,4)    | 10          | No             |

| Name                          | Description                                                                      | Data Type/Bytes | Field order | Required field |
|-------------------------------|----------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_TSF_FROM_LOC_QTY          | The quantity transferred from a source location.                                 | NUMBER(12,4)    | 11          | No             |
| F_I_TSF_FROM_LOC_COST_AMT     | The transfer cost amount for a source location in primary currency.              | NUMBER(18,4)    | 12          | No             |
| F_I_TSF_FROM_LOC_COST_AMT_LCL | The transfer cost amount for a source location in the source's local currency.   | NUMBER(18,4)    | 13          | No             |
| F_I_TSF_FROM_LOC_RTL_AMT      | The transfer retail amount for a source location in primary currency.            | NUMBER(18,4)    | 14          | No             |
| F_I_TSF_FROM_LOC_RTL_AMT_LCL  | The transfer retail amount for a source location in the source's local currency. | NUMBER(18,4)    | 15          | No             |

**ivulddm.txt**

## Business rules:

- This interface file contains unavailable inventory for an item, location combination on a given day.
- This interface file cannot contain duplicate transactions for an item\_idnt, loc\_idnt and day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name               | Description                                                                                                                                                               | Data Type/Bytes | Field order | Required field |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT          | The unique identifier of an item.                                                                                                                                         | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT           | The unique identifier of the location.                                                                                                                                    | CHARACTER(10)   | 2           | Yes            |
| DAY_DT             | The calendar day on which the transaction occurred.                                                                                                                       | DATE            | 3           | Yes            |
| F_I_UNAVL_QTY      | The quantity of the item marked as non-sellable at the location.                                                                                                          | NUMBER(12,4)    | 4           | No             |
| F_I_UNAVL_COST_AMT | The extended cost amount of unavailable inventory in primary currency. The product of the weighted average cost in primary currency and the current unavailable quantity. | NUMBER(18,4)    | 5           | No             |

| Name                   | Description                                                                                                                                                           | Data Type/Bytes | Field order | Required field |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_I_UNAVL_COST_AMT_LCL | The extended cost amount of unavailable inventory in local currency. The product of the weighted average cost in local currency and the current unavailable quantity. | NUMBER(18,4)    | 6           | No             |
| F_I_UNAVL_RTL_AMT      | The extended retail amount of unavailable inventory in primary currency. The product of the unit retail in primary currency and the current unavailable quantity.     | NUMBER(18,4)    | 7           | No             |
| F_I_UNAVL_RTL_AMT_LCL  | The extended retail amount of unavailable inventory in local currency. The product of the unit retail in local currency and the current unavailable quantity.         | NUMBER(18,4)    | 8           | No             |

| Name            | Description                                                           | Data Type/Bytes | Field order | Required field |
|-----------------|-----------------------------------------------------------------------|-----------------|-------------|----------------|
| REASN_TYPE_IDNT | The unique identifier of the reason type.                             | CHARACTER(6)    | 9           | Yes            |
| REASN_CODE_IDNT | The unique identifier of the reason code.                             | CHARACTER(6)    | 10          | Yes            |
| LOC_TYPE_CDE    | The code that indicates whether the location is a store or warehouse. | CHARACTER(2)    | 11          | Yes            |

**lptldm dm.txt**

Business rules:

- This interface file contains all the loss prevention transactions at the transaction-location-day-minute level.
- This interface file follows the fact flat file interface layout standard.

| Name            | Description                                         | Data Type/Bytes | Field order | Required field |
|-----------------|-----------------------------------------------------|-----------------|-------------|----------------|
| TRAN_IDNT       | The unique identifier of the transaction.           | VARCHAR2(30)    | 1           | Yes            |
| LOC_IDNT        | The unique identifier of the location.              | CHARACTER(10)   | 2           | Yes            |
| DAY_DT          | The calendar day on which the transaction occurred. | DATE            | 3           | Yes            |
| MIN_IDNT        | The unique identifier of the minute.                | NUMBER(4)       | 4           | Yes            |
| REASN_CODE_IDNT | The unique identifier of the reason code.           | CHARACTER(6)    | 5           | Yes            |

| Name                | Description                                                                                                                       | Data Type/Bytes | Field order | Required field |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| REASN_TYPE_IDNT     | The unique identifier of the reason type.                                                                                         | CHARACTER(6)    | 6           | Yes            |
| CSHR_IDNT           | The unique identifier for a cashier.                                                                                              | CHARACTER(10)   | 7           | Yes            |
| RGSTR_IDNT          | The unique identifier of the register.                                                                                            | CHARACTER(10)   | 8           | Yes            |
| F_LP_AMT            | The loss prevention amount, in primary currency.                                                                                  | NUMBER(18,4)    | 9           | No             |
| F_LP_AMT_LCL        | The loss prevention transaction amount, in local currency.                                                                        | NUMBER(18,4)    | 10          | No             |
| F_DISC_COUPON_COUNT | Total count of discount coupons used on one transaction. Discount coupons are issued by the store as opposed to the manufacturer. | NUMBER(16,4)    | 11          | No             |

| Name                  | Description                                                                                                                                             | Data Type/Bytes | Field order | Required field |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_DISC_COUPON_AMT     | Total amount of discount coupons used on one transaction, in primary currency. Discount coupons are issued by the store as opposed to the manufacturer. | NUMBER(18,4)    | 12          | No             |
| F_DISC_COUPON_AMT_LCL | Total amount of discount coupons used on one transaction, in local currency. Discount coupons are issued by the store as opposed to the manufacturer.   | NUMBER(18,4)    | 13          | No             |

### lptotclddm.txt

Business rules:

- This interface file contains loss prevention over/short totals.
- Amounts are summed in the target table by cshr\_idnt, rgstr\_idnt, loc\_idnt, and day\_dt.
- In each record, either rgstr\_idnt or cshr\_idnt should be filled with a value and the other field should be -1.
- This interface file follows the fact flat file interface layout standard.

| Name      | Description                          | Data Type/Bytes | Field order | Required field |
|-----------|--------------------------------------|-----------------|-------------|----------------|
| CSHR_IDNT | The unique identifier for a cashier. | CHARACTER(10)   | 1           | Yes            |



| Name                | Description                                         | Data Type/Bytes | Field order | Required field |
|---------------------|-----------------------------------------------------|-----------------|-------------|----------------|
| LOC_IDNT            | The unique identifier of the location.              | CHARACTER(10)   | 2           | Yes            |
| DAY_DT              | The calendar day on which the transaction occurred. | DATE            | 3           | Yes            |
| RGSTR_IDNT          | The unique identifier of the register.              | CHARACTER(10)   | 4           | Yes            |
| F_DRAWER_OS_AMT     | The over/short amount in primary currency.          | NUMBER(18,4)    | 5           | No             |
| F_DRAWER_OS_AMT_LCL | The over/short amount in local currency.            | NUMBER(18,4)    | 6           | No             |

**lptotlddm.txt**

Business rules:

- This interface file contains user-defined loss prevention totals.
- Amounts are summed in the target table by total type, location, and day.
- This interface file follows the fact flat file interface layout standard.

| Name            | Description                                             | Data Type/Bytes | Field order | Required field |
|-----------------|---------------------------------------------------------|-----------------|-------------|----------------|
| LOC_IDNT        | The unique identifier of the location.                  | CHARACTER(10)   | 1           | Yes            |
| DAY_DT          | The calendar day on which the transaction occurred.     | DATE            | 2           | Yes            |
| TOTAL_TYPE_IDNT | The original identifier for the total to be reconciled. | CHARACTER(10)   | 3           | Yes            |
| F_TOTAL_AMT     | The total amount in primary currency.                   | NUMBER(18,4)    | 4           | No             |
| F_TOTAL_AMT_LCL | The total amount in local currency.                     | NUMBER(18,4)    | 5           | No             |

**ncstulddm.txt**

Business rules:

- This interface file contains net cost information.
- This interface file cannot contain duplicate transactions for an item\_idnt, supp\_idnt, loc\_idnt, day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name                     | Description                                                                                                                                                                       | Data Type/Bytes | Field order | Required field |
|--------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT                | The unique identifier of an item.                                                                                                                                                 | CHARACTER(25)   | 1           | Yes            |
| SUPP_IDNT                | The unique identifier of a supplier.                                                                                                                                              | CHARACTER(10)   | 2           | Yes            |
| LOC_IDNT                 | The unique identifier of the location.                                                                                                                                            | CHARACTER(10)   | 3           | Yes            |
| DAY_DT                   | The calendar day on which the transaction occurred.                                                                                                                               | DATE            | 4           | Yes            |
| F_SUPP_BASE_COST_AMT     | The supplier base cost of the item/supplier at a given location on a given day. It is the initial cost before any deals or discounts are applied in primary currency.             | NUMBER(18,4)    | 5           | No             |
| F_SUPP_BASE_COST_AMT_LCL | The supplier base cost of the item/supplier at a given location on a given day. It is the initial cost before any deals or discounts are applied. It is stored in local currency. | NUMBER(18,4)    | 6           | No             |

| Name                    | Description                                                                                                                                                                                                                                                                                       | Data Type/Bytes | Field order | Required field |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_SUPP_NET_COST_AMT     | The supplier net cost for the item/supplier/location on a given day. It is defined as the base cost minus any deal components that are applied by the retailer. If no deals or discounts are applied at this level, the supplier net cost = supplier base cost. It is stored in primary currency. | NUMBER(18,4)    | 7           | No             |
| F_SUPP_NET_COST_AMT_LCL | The supplier net cost for the item/supplier/location on a given day. It is defined as the base cost minus any deal components that are applied by the retailer. If no deals or discounts are applied at this level, the supplier net cost = supplier base cost. It is stored in local currency.   | NUMBER(18,4)    | 8           | No             |

| Name                        | Description                                                                                                                                                                                                                                                                                                                      | Data Type/Bytes | Field order | Required field |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_SUPP_NET_NET_COST_AMT     | The supplier net net cost of the item/supplier/location on a given day. It is defined as the net cost minus any deal components designated by a retailer as applicable to the net net cost. If no deals or discounts are applied at this level, the supplier net net cost = supplier net cost. It is stored in primary currency. | NUMBER(18,4)    | 9           | No             |
| F_SUPP_NET_NET_COST_AMT_LCL | The supplier net net cost of the item/supplier/location on a given day. It is defined as the net cost minus any deal components designated by a retailer as applicable to the net net cost. If no deals or discounts are applied at this level, the supplier net net cost = supplier net cost. It is stored in local currency.   | NUMBER(18,4)    | 10          | No             |

| Name                         | Description                                                                                                                                                                                                                                                                                                                                                                                                | Data Type/Bytes | Field order | Required field |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_SUPP_DEAD_NET_COST_AMT     | The supplier dead net cost of the item/supplier/location on a given day. It is the final cost after all deals or discounts are applied. It is defined as the net net cost minus any deal components designated by a retailer as applicable to the dead net cost. If no deals or discounts are applied at this level, the supplier dead net cost = supplier net net cost. It is stored in primary currency. | NUMBER(18,4)    | 11          | No             |
| F_SUPP_DEAD_NET_COST_AMT_LCL | The supplier dead net cost of the item/supplier/location on a given day. It is the final cost after all deals or discounts are applied. It is defined as the net net cost minus any deal components designated by a retailer as applicable to the dead net cost. If no deals or discounts are applied at this level, the supplier dead net cost = supplier net net cost. It is stored in local currency.   | NUMBER(18,4)    | 12          | No             |

**orgaradm.txt**

Business rules:

- This interface file contains areas within a chain.
- This interface file cannot contain duplicate records for an area\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name          | Description                                                         | Data Type/Bytes | Field order | Required field |
|---------------|---------------------------------------------------------------------|-----------------|-------------|----------------|
| AREA_IDNT     | The unique identifier of an area in the organizational hierarchy.   | VARCHAR2(10)    | 1           | Yes            |
| AREA_DESC     | The name of the area in the organizational hierarchy.               | VARCHAR2(120)   | 2           | No             |
| AREA_MGR_NAME | The name of the manager for the area.                               | VARCHAR2(120)   | 3           | No             |
| CHAIN_IDNT    | The unique identifier of the chain in the organizational hierarchy. | VARCHAR2(10)    | 4           | Yes            |

### orgchandm.txt

Business rules:

- This interface file contains channels within a company.
- This interface file cannot contain duplicate records for a channel\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name         | Description                                                                                                                                                                                                                       | Data Type/Bytes | Field order | Required field |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| CHANNEL_IDNT | The unique identifier of the channel in the organizational hierarchy.                                                                                                                                                             | VARCHAR2(4)     | 1           | Yes            |
| BANNER_IDNT  | The unique identifier of a banner. Banner represents the name of a retail company's subsidiary that is recognizable to the consumer or the name of the store as it appears on the catalog, web channel or brick and mortar store. | VARCHAR2(4)     | 2           | Yes            |

| Name         | Description              | Data Type/Bytes | Field order | Required field |
|--------------|--------------------------|-----------------|-------------|----------------|
| CHANNEL_TYPE | The type of channel.     | VARCHAR2(6)     | 3           | No             |
| CHANNEL_DESC | The name of the channel. | VARCHAR2(120)   | 4           | No             |
| BANNER_DESC  | The name of the banner.  | VARCHAR2(120)   | 5           | No             |

**orgchndm.txt**

Business rules:

- This interface file contains chains within a company.
- This interface file cannot contain duplicate records for a chain\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name           | Description                                                                 | Data Type/Bytes | Field order | Required field |
|----------------|-----------------------------------------------------------------------------|-----------------|-------------|----------------|
| CHAIN_IDNT     | The unique identifier of the chain in the organizational hierarchy.         | VARCHAR2(10)    | 1           | Yes            |
| CMPY_IDNT      | The unique identifier of the company in product and organization hierarchy. | VARCHAR2(4)     | 2           | Yes            |
| CHAIN_DESC     | The name of the chain in the organizational hierarchy.                      | VARCHAR2(120)   | 3           | No             |
| CHAIN_MGR_NAME | The name of the manager for the chain.                                      | VARCHAR2(120)   | 4           | No             |

**orgdisdm.txt**

Business rules:

- This interface file contains districts within a region.
- This interface file cannot contain duplicate records for a distt\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name       | Description                                                        | Data Type/Bytes | Field order | Required field |
|------------|--------------------------------------------------------------------|-----------------|-------------|----------------|
| DISTT_IDNT | The unique identifier of a district in the organization hierarchy. | VARCHAR2(10)    | 1           | Yes            |

| Name           | Description                                                        | Data Type/Bytes | Field order | Required field |
|----------------|--------------------------------------------------------------------|-----------------|-------------|----------------|
| DISTT_DESC     | The name of the district in the organization hierarchy.            | VARCHAR2(120)   | 2           | No             |
| DISTT_MGR_NAME | The name of the manager responsible for this district.             | VARCHAR2(120)   | 3           | No             |
| REGN_IDNT      | The unique identifier of the region in the organization hierarchy. | VARCHAR2(10)    | 4           | Yes            |

### orglmdm.txt

Business rules:

- This interface file defines the associations between location and location list.
- This interface file cannot contain duplicate records for a loclst\_idnt, loc\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name         | Description                                                           | Data Type/Bytes | Field order | Required field |
|--------------|-----------------------------------------------------------------------|-----------------|-------------|----------------|
| LOCLST_IDNT  | The unique identifier of a location list.                             | VARCHAR2(10)    | 1           | Yes            |
| LOC_IDNT     | The unique identifier of the location.                                | VARCHAR2(10)    | 2           | Yes            |
| LOC_TYPE_CDE | The code that indicates whether the location is a store or warehouse. | VARCHAR2(2)     | 3           | Yes            |

### orglocdm.txt

Business rules:

- This interface file contains locations within a district.
- This interface file cannot contain duplicate records for a loc\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name     | Description                            | Data Type/Bytes | Field order | Required field |
|----------|----------------------------------------|-----------------|-------------|----------------|
| LOC_IDNT | The unique identifier of the location. | VARCHAR2(10)    | 1           | Yes            |



| Name           | Description                                                                                        | Data Type/Bytes | Field order | Required field |
|----------------|----------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_TYPE_CDE   | The code that indicates whether the location is a store or warehouse.                              | VARCHAR2(2)     | 2           | Yes            |
| LOC_DESC       | The description or name of the store or warehouse.                                                 | VARCHAR2(240)   | 3           | No             |
| LOC_DESC_10    | The 10 character abbreviation of the store name.                                                   | VARCHAR2(10)    | 4           | No             |
| LOC_DESC_3     | The 3 character abbreviation of the store name.                                                    | VARCHAR2(3)     | 5           | No             |
| LOC_SECND_DESC | The secondary description or name of the store or warehouse.                                       | VARCHAR2(240)   | 6           | No             |
| LOC_TYPE_DESC  | The description of the loc_type_cde that indicates whether the location is a store or warehouse. . | VARCHAR2(120)   | 7           | No             |
| DISTT_IDNT     | The unique identifier of a district in the organization hierarchy.                                 | VARCHAR2(10)    | 8           | Yes            |

| Name            | Description                                                                                                 | Data Type/Bytes | Field order | Required field |
|-----------------|-------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| DISTT_DESC      | The name of the district in the organization hierarchy.                                                     | VARCHAR2(120)   | 9           | No             |
| CRNCY_CDE_IDNT  | The unique identifier of the currency code.                                                                 | VARCHAR2(10)    | 10          | No             |
| CRNCY_CDE_DESC  | The description of local currency code. That is, description for USD = US Dollar.                           | VARCHAR2(120)   | 11          | No             |
| PHY_WH_IDNT     | The unique identifier of the physical warehouse that is assigned to the virtual warehouse.                  | VARCHAR2(10)    | 12          | No             |
| VIRTUAL_WH_IDNT | The identifier of the virtual warehouse.                                                                    | VARCHAR2(10)    | 13          | No             |
| STOCKHOLD_IND   | Indicates whether the location can hold stock. In a non-multichannel environment this stores always be "Y." | VARCHAR2(1)     | 14          | No             |
| CHANNEL_IDNT    | The unique identifier of the channel in the organizational hierarchy.                                       | VARCHAR2(4)     | 15          | No             |

| Name                | Description                                                                                                                                                                                                                       | Data Type/Bytes | Field order | Required field |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| CHANNEL_DESC        | The name of the channel.                                                                                                                                                                                                          | VARCHAR2(120)   | 16          | No             |
| BANNER_IDNT         | The unique identifier of a banner. Banner represents the name of a retail company's subsidiary that is recognizable to the consumer or the name of the store as it appears on the catalog, web channel or brick and mortar store. | VARCHAR2(4)     | 17          | No             |
| BANNER_DESC         | The name of the banner.                                                                                                                                                                                                           | VARCHAR2(120)   | 18          | No             |
| LOC_ADDR            | The street address of the store or warehouse.                                                                                                                                                                                     | VARCHAR2(722)   | 19          | No             |
| LOC_CITY_NAME       | The city in which the store or warehouse is located.                                                                                                                                                                              | VARCHAR2(120)   | 20          | No             |
| LOC_ST_OR_PRVNC_CDE | The state or province code in which the store or warehouse is located.                                                                                                                                                            | VARCHAR2(7)     | 21          | No             |
| LOC_CNTRY_CDE       | The country code in which the store or warehouse is located.                                                                                                                                                                      | VARCHAR2(10)    | 22          | No             |

| Name                    | Description                                                                                                         | Data Type/Bytes | Field order | Required field |
|-------------------------|---------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_CNTRY_DESC          | The description or name of the country code in which the store or warehouse is located.                             | VARCHAR2(120)   | 23          | No             |
| LOC_PSTL_CDE            | The postal code of the store or warehouse.                                                                          | VARCHAR2(30)    | 24          | No             |
| LOC_MGR_NAME            | The name of the manager responsible for this store. Only valid for the store Locations.                             | VARCHAR2(120)   | 25          | No             |
| LOC_FMT_CDE             | The code that indicates the type of format of the location. Only valid for store locations.                         | VARCHAR2(5)     | 26          | No             |
| LOC_SELLING_AREA        | The location's total selling area.                                                                                  | NUMBER(8)       | 27          | No             |
| LOC_TOT_LINEAR_DISTANCE | The total linear selling space of the location.                                                                     | NUMBER(8)       | 28          | No             |
| LOC_PRMTN_ZNE_CDE       | The code that indicates the promotion zone for which this location is a member. Only valid for the store Locations. | VARCHAR2(5)     | 29          | No             |

| Name                | Description                                                                                                           | Data Type/Bytes | Field order | Required field |
|---------------------|-----------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_TRNSFR_ZNE_CDE  | The code that indicates the transfer zone for which this location is a member. Only valid for the store locations.    | VARCHAR2(5)     | 30          | No             |
| LOC_VAT_REGN        | The number of the Value Added Tax region in which this store or warehouse is contained.                               | NUMBER(4)       | 31          | No             |
| LOC_VAT_INCLUDE_IND | Indicates whether or not Value Added Tax is included in the retail prices for the store. Valid values are 'Y' or 'N'. | VARCHAR2(1)     | 32          | No             |
| LOC_MALL_NAME       | The name of the mall in which the store is located.                                                                   | VARCHAR2(120)   | 33          | No             |

| Name              | Description                                                                                                                                                              | Data Type/Bytes | Field order | Required field |
|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_DEFAULT_WH    | The number of the warehouse that may be used as the default for creating cross-dock masks. This determines which stores are associated with or sourced from a warehouse. | VARCHAR2(10)    | 34          | No             |
| LOC_BREAK_PAC_IND | Indicates whether or not the warehouse is capable of distributing less than the supplier case quantity. Valid values are 'Y' or 'N'.                                     | VARCHAR2(1)     | 35          | No             |
| LOC_REMODEL_DT    | The date on which the store was last remodeled.                                                                                                                          | DATE            | 36          | No             |
| LOC_START_DT      | The start date for location.                                                                                                                                             | DATE            | 37          | No             |
| LOC_END_DT        | The end date for a location.                                                                                                                                             | DATE            | 38          | No             |
| LOC_TOT_AREA      | The total area of the location.                                                                                                                                          | NUMBER(8)       | 39          | No             |

| Name                | Description                                                                                                       | Data Type/Bytes | Field order | Required field |
|---------------------|-------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_NO_LOAD_DOCKS   | This field is client specific. The definition and use of this field is customizable for each client.              | VARCHAR2(4)     | 40          | No             |
| LOC_NO_UNLOAD_DOCKS | This field is client specific. The definition and use of this field is customizable for each client.              | VARCHAR2(4)     | 41          | No             |
| LOC_UPS_DISTT       | The code that indicates the UPS district for which this location is a member. Only valid for the store locations. | NUMBER(2)       | 42          | No             |
| LOC_TIME_ZNE        | The code that indicates the time zone for which this location is a member. Only valid for the store locations.    | VARCHAR2(10)    | 43          | No             |
| LOC_FASH_LINE_NO    | This field is client specific. The definition and use of this field is customizable for each client.              | VARCHAR2(9)     | 44          | No             |

| <b>Name</b>       | <b>Description</b>                                                                                   | <b>Data Type/Bytes</b> | <b>Field order</b> | <b>Required field</b> |
|-------------------|------------------------------------------------------------------------------------------------------|------------------------|--------------------|-----------------------|
| LOC_COMP_CDE      | This field is client specific. The definition and use of this field is customizable for each client. | VARCHAR2(2)            | 45                 | No                    |
| LOC_STORE_VOL_CAT | This field is client specific. The definition and use of this field is customizable for each client. | VARCHAR2(2)            | 46                 | No                    |
| LOC_PAY_CAT       | This field is client specific. The definition and use of this field is customizable for each client. | VARCHAR2(1)            | 47                 | No                    |
| LOC_ACCT_CLK_ID   | This field is client specific. The definition and use of this field is customizable for each client. | CHARACTER(3)           | 48                 | No                    |



| Name                 | Description                                                                                               | Data Type/Bytes | Field order | Required field |
|----------------------|-----------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_FMT_DESC         | The description or name of the location format code of this location. Only valid for the store locations. | CHARACTER(120)  | 49          | No             |
| LOC_ST_OR_PRVNC_DESC | The description or name of the state or province in which the store or warehouse is located.              | VARCHAR2(120)   | 50          | No             |
| LOC_TRNSFR_ZNE_DESC  | The description or name of the transfer zone code of this location. Only valid for the store locations.   | VARCHAR2(120)   | 51          | No             |
| LOC_PRMTN_ZNE_DESC   | The description or name of the promotion zone code of this location. Only valid for the store locations.  | CHARACTER(120)  | 52          | No             |
| STORE_CLASS          | This value is populated for RPAS only. Null if RPAS is not used.                                          | CHARACTER(1)    | 53          | No             |

| Name             | Description                                                      | Data Type/Bytes | Field order | Required field |
|------------------|------------------------------------------------------------------|-----------------|-------------|----------------|
| START_ORDER_DAYS | This value is populated for RPAS only. Null if RPAS is not used. | CHARACTER(3)    | 54          | No             |
| FORECAST_WH_IND  | This value is populated for RPAS only. Null if RPAS is not used. | CHARACTER(1)    | 55          | No             |

### **orgloldm.txt**

Business rules:

- This interface file contains one record for each location list. A location list is normally used to group locations for reporting purposes.
- This interface file cannot contain duplicate records for a loclst\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name        | Description                                                     | Data Type/Bytes | Field order | Required field |
|-------------|-----------------------------------------------------------------|-----------------|-------------|----------------|
| LOCLST_IDNT | The unique identifier of a location list.                       | VARCHAR2(10)    | 1           | Yes            |
| CREATE_ID   | The login ID of the person who created the location list.       | VARCHAR2(30)    | 2           | Yes            |
| LOCLST_DESC | The description or name of the location list unique identifier. | VARCHAR2(120)   | 3           | No             |

**orgltdm.txt**

Business rules:

- This interface file defines the associations between location and location traits.
- This interface file cannot contain duplicate records for a loc\_trait\_idnt, loc\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name           | Description                                                                           | Data Type/Bytes | Field order | Required field |
|----------------|---------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_TRAIT_IDNT | The location trait unique identifier. Only valid entries are for the store locations. | VARCHAR2(10)    | 1           | Yes            |
| LOC_IDNT       | The unique identifier of the location.                                                | VARCHAR2(10)    | 2           | Yes            |
| LOC_TYPE_CDE   | The code that indicates whether the location is a store or warehouse.                 | VARCHAR2(2)     | 3           | No             |

**orgltrdm.txt**

Business rules:

- This interface file cannot contain duplicate records for a loc\_trait\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name           | Description                                                                           | Data Type/Bytes | Field order | Required field |
|----------------|---------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_TRAIT_IDNT | The location trait unique identifier. Only valid entries are for the store locations. | VARCHAR2(10)    | 1           | Yes            |
| LOC_TRAIT_DESC | The description or name of the location trait unique identifier.                      | VARCHAR2(120)   | 2           | No             |

**orgrgndm.txt**

Business rules:

- This interface file contains regions within an area.
- This interface file cannot contain duplicate records for a regn\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name          | Description                                                          | Data Type/Bytes | Field order | Required field |
|---------------|----------------------------------------------------------------------|-----------------|-------------|----------------|
| REGN_IDNT     | The unique identifier of the region in the organization hierarchy.   | VARCHAR2(10)    | 1           | Yes            |
| REGN_DESC     | The description or name of the region in the organization hierarchy. | VARCHAR2(120)   | 2           | No             |
| REGN_MGR_NAME | The name of the manager for the region.                              | VARCHAR2(120)   | 3           | No             |
| AREA_IDNT     | The unique identifier of an area in the organizational hierarchy.    | VARCHAR2(10)    | 4           | Yes            |

**phasdm.txt**

Business rules:

- This interface file contains phases. Phases are periods of time within a season. Each day should fall within no more than one phase.
- This interface file cannot contain duplicate records for a phase\_idnt, seasn\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name           | Description                            | Data Type/Bytes | Field order | Required field |
|----------------|----------------------------------------|-----------------|-------------|----------------|
| SEASN_IDNT     | The season identifier.                 | VARCHAR2(3)     | 1           | Yes            |
| PHASE_IDNT     | The unique identifier of the phase.    | VARCHAR2(3)     | 2           | Yes            |
| PHASE_START_DT | The beginning date of the phase.       | DATE            | 3           | Yes            |
| PHASE_END_DT   | The ending date of the phase.          | DATE            | 4           | Yes            |
| PHASE_DESC     | The description or name for the phase. | VARCHAR2(120)   | 5           | No             |

**prcilddm.txt**

Business rules:

- This interface file contains prices by the tracking level item and location combination on a given day.
- This interface file cannot contain duplicate transactions for an item\_idnt, loc\_idnt, day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name             | Description                                                           | Data Type/Bytes | Field order | Required field |
|------------------|-----------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT        | The unique identifier of an item.                                     | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT         | The unique identifier of the location.                                | CHARACTER(10)   | 2           | Yes            |
| DAY_DT           | The calendar day on which the transaction occurred.                   | DATE            | 3           | Yes            |
| LOC_TYPE_CDE     | The code that indicates whether the location is a store or warehouse. | CHARACTER(2)    | 4           | Yes            |
| CHNG_CDE         | The reason code for price change.                                     | VARCHAR2(2)     | 5           | No             |
| F_MULTI_UNIT_QTY | The number of units that comprise a multi-unit transaction.           | NUMBER(12,4)    | 6           | No             |
| F_UNIT_RTL_AMT   | The unit value of new retail valuation/price in primary currency.     | NUMBER(18,4)    | 7           | No             |

| Name                     | Description                                                                                          | Data Type/Bytes | Field order | Required field |
|--------------------------|------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_UNIT_RTL_AMT_LCL       | The unit value of new retail valuation/price in local currency.                                      | NUMBER(18,4)    | 8           | No             |
| F_MULTI_UNIT_RTL_AMT     | The unit dollar value of new retail multi unit valuation/price.                                      | NUMBER(18,4)    | 9           | No             |
| F_MULTI_UNIT_RTL_AMT_LCL | The unit dollar value of new retail multi unit valuation/price in local currency.                    | NUMBER(18,4)    | 10          | No             |
| SELLING_UOM_CDE          | The selling unit of measure code for an item's single-unit retail. This is a non-aggregatable value. | VARCHAR2(4)     | 11          | No             |
| MULTI_SELLING_UOM_CDE    | The selling unit of measure code for an item's multi-unit retail. This is a non-aggregatable value.  | VARCHAR2(4)     | 12          | No             |

**prdcldsm.txt**

Business rules:

- This interface file contains classes within a department.
- This interface file cannot contain duplicate records for a dept\_idnt, class\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name            | Description                                                     | Data Type/Bytes | Field order | Required field |
|-----------------|-----------------------------------------------------------------|-----------------|-------------|----------------|
| CLASS_IDNT      | The unique identifier of the class in the product hierarchy.    | VARCHAR2(4)     | 1           | Yes            |
| DEPT_IDNT       | The unique identifier of a department in the product hierarchy. | VARCHAR2(4)     | 2           | Yes            |
| CLASS_DESC      | The name of the class in the product hierarchy.                 | VARCHAR2(120)   | 3           | No             |
| CLASS_BUYR_IDNT | The unique identifier for the buyer of the class.               | VARCHAR2(4)     | 4           | No             |
| CLASS_BUYR_NAME | The name of the buyer for this class of products                | VARCHAR2(120)   | 5           | No             |
| CLASS_MRCH_IDNT | The unique identifier of the merchandiser for this department.  | VARCHAR2(4)     | 6           | No             |
| CLASS_MRCH_NAME | The name of the merchandiser for this class of products.        | VARCHAR2(120)   | 7           | No             |

### prdcmpdm.txt

Business rules:

- This interface file contains company information.
- This interface file cannot contain duplicate records for a cmpy\_idnt.
- This interface file follows the dimension flat file interface layout standard.

| Name      | Description                                                                 | Data Type/Bytes | Field order | Required field |
|-----------|-----------------------------------------------------------------------------|-----------------|-------------|----------------|
| CMPY_IDNT | The unique identifier of the company in product and organization hierarchy. | VARCHAR2(4)     | 1           | Yes            |
| CMPY_DESC | The name of the company in product and organization hierarchy.              | VARCHAR2(120)   | 2           | No             |

### prddepdmm.txt

Business rules:

- This interface file contains departments within a group.
- This interface file cannot contain duplicate records for a dept\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name           | Description                                                     | Data Type/Bytes | Field order | Required field |
|----------------|-----------------------------------------------------------------|-----------------|-------------|----------------|
| DEPT_IDNT      | The unique identifier of a department in the product hierarchy. | VARCHAR2(4)     | 1           | Yes            |
| GRP_IDNT       | The unique identifier of the group in the product hierarchy.    | VARCHAR2(4)     | 2           | Yes            |
| DEPT_DESC      | The name of the department in the product hierarchy.            | VARCHAR2(120)   | 3           | No             |
| DEPT_BUYR_IDNT | The unique identifier of the buyer for the department.          | VARCHAR2(4)     | 4           | No             |



| Name                | Description                                                                                                             | Data Type/Bytes | Field order | Required field |
|---------------------|-------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| DEPT_BUYR_NAME      | The name of the buyer which corresponds to the dept_buyr_idnt for the department.                                       | VARCHAR2(120)   | 5           | No             |
| DEPT_MRCH_IDNT      | The unique character representation of the merchandiser for the department.                                             | VARCHAR2(4)     | 6           | No             |
| DEPT_MRCH_NAME      | The name of the merchandiser that corresponds to the dept_mrch_idnt for the department.                                 | VARCHAR2(120)   | 7           | No             |
| PRFT_CALC_TYPE_CDE  | The unique code which determines whether profit is calculated based on cost or retail for the department.               | VARCHAR2(1)     | 8           | No             |
| PRFT_CALC_TYPE_DESC | The description of the what method the profit was calculated for the department. Typically, it would be cost or retail. | VARCHAR2(120)   | 9           | No             |

| Name               | Description                                                                                                                      | Data Type/Bytes | Field order | Required field |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| PURCH_TYPE_CDE     | The code that determines which type of stock the items are within this department (that is, normal stock vs. consignment stock). | VARCHAR2(1)     | 10          | No             |
| PURCH_TYPE_DESC    | The description of the type of merchandise within the department (that is, normal stock, consignment stock, etc.).               | VARCHAR2(120)   | 11          | No             |
| BUD_INT            | The budgeted intake percentage. The term is synonymous with markup percent of retail.                                            | NUMBER(12,4)    | 12          | No             |
| BUD_MKUP           | The budgeted markup percentage. This term is synonymous with markup percent of cost.                                             | NUMBER(12,4)    | 13          | No             |
| TOTL_MKT_AMT       | The total market amount expected for this department.                                                                            | NUMBER(18,4)    | 14          | No             |
| MKUP_CALC_TYPE_CDE | The code which determines how markup is calculated for the department.                                                           | VARCHAR2(1)     | 15          | No             |

| Name                | Description                                                                                  | Data Type/Bytes | Field order | Required field |
|---------------------|----------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| MKUP_CALC_TYPE_DESC | The description of the how the markup is calculated for the department.                      | VARCHAR2(120)   | 16          | No             |
| OTB_CALC_TYPE_CDE   | The code that determines if Open To Buy (OTB) is based on cost or retail for the department. | VARCHAR2(1)     | 17          | No             |
| OTB_CALC_TYPE_DESC  | The description of the whether the OTB is calculated based on cost or retail.                | VARCHAR2(120)   | 18          | No             |

**prddiffdm.txt**

Business rules:

- This interface file contains all item differentiator identifiers, along with their associated NRF industry codes.
- This interface file cannot contain duplicate records for a diff\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name              | Description                                                                                                                                                                 | Data Type/Bytes | Field order | Required field |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| DIFF_IDNT         | The uniquely identifier of a differentiator. (for example, diff_type = 'S' might have these differentiators: 1, 50, 1000; then diff_type = 'C' cannot use the same numbers) | VARCHAR2(10)    | 1           | Yes            |
| DIFF_TYPE         | The unique identifier of a differentiator type. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern).                                           | CHARACTER(6)    | 2           | No             |
| DIFF_DESC         | The description of the differentiator                                                                                                                                       | VARCHAR2(120)   | 3           | No             |
| INDUSTRY_CDE      | A unique number that represents all possible combinations of sizes.                                                                                                         | VARCHAR2(10)    | 4           | No             |
| INDUSTRY_SUBGROUP | A unique number that represents all different color range group.                                                                                                            | VARCHAR2(10)    | 5           | No             |

**prddivdm.txt**

Business rules:

- This interface file contains divisions within a company.
- This interface file cannot contain duplicate records for a div\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name          | Description                                                                 | Data Type/Bytes | Field order | Required field |
|---------------|-----------------------------------------------------------------------------|-----------------|-------------|----------------|
| DIV_IDNT      | The unique identifier of a division in the product hierarchy.               | VARCHAR2(4)     | 1           | Yes            |
| CMPY_IDNT     | The unique identifier of the company in product and organization hierarchy. | VARCHAR2(4)     | 2           | Yes            |
| DIV_DESC      | The name of the division in the product hierarchy.                          | VARCHAR2(120)   | 3           | No             |
| DIV_BUYR_IDNT | The unique character representation of the buyer for the division.          | VARCHAR2(4)     | 4           | No             |
| DIV_BUYR_NAME | The name of the buyer for the division.                                     | VARCHAR2(120)   | 5           | No             |
| DIV_MRCH_IDNT | The unique identifier of the merchandiser for the division.                 | VARCHAR2(4)     | 6           | No             |
| DIV_MRCH_NAME | The name of the merchandiser for the division.                              | VARCHAR2(120)   | 7           | No             |

**prddtypdm.txt**

Business rules:

- This interface file contains differentiator (diff) types.
- This interface file cannot contain duplicate records for a diff\_type.

| Name           | Description                                                                                                                          | Data Type/Bytes | Field order | Required field |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| DIFF_TYPE      | The unique identifier of a differentiator type.<br>(for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). | VARCHAR2(6)     | 1           | Yes            |
| DIFF_TYPE_DESC | The description of the differentiator type.                                                                                          | VARCHAR2(120)   | 2           | Yes            |

**prdgrpdm.txt**

Business rules:

- This interface file contains groups within a division.
- This interface file cannot contain duplicate records for a grp\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name          | Description                                                     | Data Type/Bytes | Field order | Required field |
|---------------|-----------------------------------------------------------------|-----------------|-------------|----------------|
| GRP_IDNT      | The unique identifier of the group in the product hierarchy.    | VARCHAR2(4)     | 1           | Yes            |
| DIV_IDNT      | The unique identifier of a division in the product hierarchy.   | VARCHAR2(4)     | 2           | Yes            |
| GRP_DESC      | The name of the group in the product hierarchy.                 | VARCHAR2(120)   | 3           | No             |
| GRP_BUYR_IDNT | The unique character representation of the buyer for the group. | VARCHAR2(4)     | 4           | No             |

| Name          | Description                                                                       | Data Type/Bytes | Field order | Required field |
|---------------|-----------------------------------------------------------------------------------|-----------------|-------------|----------------|
| GRP_BUYR_NAME | The name of the buyer that corresponds with the buyr_idnt for the group.          | VARCHAR2(120)   | 5           | No             |
| GRP_MRCH_IDNT | The unique identifier of the merchandiser for the group.                          | VARCHAR2(4)     | 6           | No             |
| GRP_MRCH_NAME | The name of the merchandiser that corresponds to the grp_mrch_idnt for the group. | VARCHAR2(120)   | 7           | No             |

**prdisldm.txt**

Business rules:

- This interface file contains records associating tracking level items with locations and primary suppliers.
- This interface file cannot contain duplicate records for a supp\_idnt, item\_idnt, loc\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name         | Description                              | Data Type/Bytes | Field order | Required field |
|--------------|------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT    | The unique identifier of an item.        | VARCHAR2(25)    | 1           | Yes            |
| SUPP_IDNT    | The unique identifier of a supplier.     | VARCHAR2(10)    | 2           | Yes            |
| LOC_IDNT     | The unique identifier of the location.   | VARCHAR2(10)    | 3           | Yes            |
| SUPP_PRT_NBR | The corresponding suppliers part number. | VARCHAR2(30)    | 4           | No             |

| Name                | Description                                                                                                                              | Data Type/Bytes | Field order | Required field |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| PRMY_SUPP_IND       | Indicator to maintain and track the primary supplier for an item. Y indicates this is a primary supplier for the item at the location.   | VARCHAR2(1)     | 5           | No             |
| PRESENTATION_METHOD | The description of the packaging (if any) being taken into consideration in the specified dimensions. Valid values are 'JHOOK', 'STACK'. | VARCHAR2(6)     | 6           | No             |
| F_SUPP_CASE_QTY     | The quantity of the item in an orderable case pack from the primary supplier.                                                            | NUMBER(12,4)    | 7           | No             |

**prditmdm.txt**

## Business rules:

- This interface file contains items within a subclass, class, and department. The combination of subclass, class and department makes an item unique. For example, item 100 cannot be identified by subclass 10, because subclass 10 can belong to different classes, and represent 2 different subclasses. Item 100 belongs to a combination of subclass, class and department.
- This interface file cannot contain duplicate records for an item\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.
- This interface file only contains approved items (STATUS = 'A').



| Name        | Description                                                                                                           | Data Type/Bytes | Field order | Required field |
|-------------|-----------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT   | The unique identifier of an item.                                                                                     | VARCHAR2(25)    | 1           | Yes            |
| LEVEL1_IDNT | The unique identifier of the first level item of the family.                                                          | VARCHAR2(25)    | 2           | No             |
| LEVEL2_IDNT | The unique identifier of the second level item of the family.                                                         | VARCHAR2(25)    | 3           | No             |
| LEVEL3_IDNT | The unique identifier of the third level item of the family.                                                          | VARCHAR2(25)    | 4           | No             |
| ITEM_LEVEL  | The number indicating which of the three levels the item resides. Valid values are 1, 2 and 3.                        | NUMBER(1)       | 5           | Yes            |
| TRAN_LEVEL  | The number indicating which of the three levels transactions occur for the item's group. Valid values are 1, 2 and 3. | NUMBER(1)       | 6           | Yes            |
| DIFF_1      | One of the four differentiator identifier available from the source system.                                           | CHARACTER(10)   | 7           | No             |

| <b>Name</b>          | <b>Description</b>                                                          | <b>Data Type/Bytes</b> | <b>Field order</b> | <b>Required field</b> |
|----------------------|-----------------------------------------------------------------------------|------------------------|--------------------|-----------------------|
| DIFF_2               | One of the four differentiator identifier available from the source system. | CHARACTER(10)          | 8                  | No                    |
| DIFF_3               | One of the four differentiator identifier available from the source system. | CHARACTER(10)          | 9                  | No                    |
| DIFF_4               | One of the four differentiator identifier available from the source system. | CHARACTER(10)          | 10                 | No                    |
| ITEM_AGGREGATE_IND   | This value is populated for RPAS only. Null if RPAS is not used.            | CHARACTER(1)           | 11                 | No                    |
| DIFF_1_AGGREGATE_IND | This value is populated for RPAS only. Null if RPAS is not used.            | CHARACTER(1)           | 12                 | No                    |
| DIFF_2_AGGREGATE_IND | This value is populated for RPAS only. Null if RPAS is not used.            | CHARACTER(1)           | 13                 | No                    |
| DIFF_3_AGGREGATE_IND | This value is populated for RPAS only. Null if RPAS is not used.            | CHARACTER(1)           | 14                 | No                    |

| Name                 | Description                                                                                                                                                                                                                          | Data Type/Bytes | Field order | Required field |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| DIFF_4_AGGREGATE_IND | This value is populated for RPAS only. Null if RPAS is not used.                                                                                                                                                                     | CHARACTER(1)    | 15          | No             |
| PACK_IND             | Indicates if the item is a pack.                                                                                                                                                                                                     | CHARACTER(1)    | 16          | No             |
| PACK_SELLABLE_CDE    | Indicates whether the pack is sellable. A sellable pack is a group of items that is to be sold as one item, whether the pack arrived as orderable or if the retailer took it upon themselves to package and sell the items together. | VARCHAR2(6)     | 17          | No             |
| PACK_SELLABLE_DESC   | The pack sellable description. Valid descriptions are: Sellable, Non-sellable.                                                                                                                                                       | VARCHAR2(120)   | 18          | No             |
| PACK_SIMPLE_CDE      | Indicates whether the pack is simple. A simple pack is the grouping of multiples of one particular item to be sold as one item. An example would be a twelve pack of cola.                                                           | VARCHAR2(6)     | 19          | No             |

| Name                | Description                                                                                                                                                                                                                    | Data Type/Bytes | Field order | Required field |
|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| PACK_SIMPLE_DESC    | The pack simple description. Valid descriptions are: Simple, complex.                                                                                                                                                          | VARCHAR2(120)   | 20          | No             |
| PACK_ORDERABLE_CDE  | The abbreviated code for the pack order type: vendor or buyer. An orderable pack is a pack whose contents are specified by the buyer. A vendor pack is a pack that is packaged by the vendor and can only be ordered that way. | VARCHAR2(6)     | 21          | No             |
| PACK_ORDERABLE_DESC | The pack order type description.                                                                                                                                                                                               | VARCHAR2(120)   | 22          | No             |
| PACK_IND            | Indicates if the item is a pack.                                                                                                                                                                                               | VARCHAR2(1)     | 16          | No             |
| PACKAGE_UOM         | The unit of measure associated with the package size.                                                                                                                                                                          | VARCHAR2(4)     | 23          | No             |
| PACKAGE_SIZE        | The size of the product printed on any packaging.                                                                                                                                                                              | NUMBER(12,4)    | 24          | No             |

| Name            | Description                                                                                                               | Data Type/Bytes | Field order | Required field |
|-----------------|---------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| SBCLASS_IDNT    | The unique identifier of the subclass in the product hierarchy.                                                           | VARCHAR2(4)     | 25          | Yes            |
| CLASS_IDNT      | The unique identifier of the class in the product hierarchy.                                                              | VARCHAR2(4)     | 26          | Yes            |
| DEPT_IDNT       | The unique identifier of a department in the product hierarchy.                                                           | VARCHAR2(4)     | 27          | Yes            |
| ITEM_DESC       | The long description of the item. This description is used through out the system to help online users identify the item. | VARCHAR2(255)   | 28          | No             |
| ITEM_SECND_DESC | The secondary description of the item.                                                                                    | VARCHAR2(255)   | 29          | No             |
| ITEM_SHRT_DESC  | The shortened description of the item. This description may be the default for downloading to the point of sale system.   | VARCHAR2(120)   | 30          | No             |

| Name               | Description                                                                                                                                                  | Data Type/Bytes | Field order | Required field |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_NBR_TYPE_CDE  | The code specifying what type the item is. Some valid values for this field are ITEM, UPC-A, EAN13, ISBN, etc.                                               | VARCHAR2(6)     | 31          | No             |
| ITEM_NBR_TYPE_DESC | The description of the item number type.                                                                                                                     | VARCHAR2(120)   | 32          | No             |
| STND_UOM_CDE       | The string that uniquely identifies the unit of measure.                                                                                                     | VARCHAR2(6)     | 33          | No             |
| STND_UOM_DESC      | The description of the UOM_CDE for clarity.                                                                                                                  | CHARACTER(120)  | 34,         | No             |
| FORECAST_IND       | This value is populated for RPAS only. Null if RPAS is not used.                                                                                             | CHARACTER(1)    | 35          | Yes            |
| SELLABLE_IND       | Indicates whether the item can be sold. If 'N', then the only analysis available is on customer order lines of type partial within Customer Order Management | VARCHAR2(1)     | 36          | No             |

| Name            | Description                                                                                               | Data Type/Bytes | Field order | Required field |
|-----------------|-----------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| INV_IND         | Indicates whether an item is an inventory item or a non-inventory item (such as gift certificates, labor) | VARCHAR2(1)     | 37          | No             |
| MRCH_IND        | Indicates whether the item's sales are financially tracked in the stock ledger.                           | VARCHAR2(1)     | 38          | No             |
| RECIPE_CARD_IND | Indicates whether a recipe card is available for the item.                                                | VARCHAR2(1)     | 39          | No             |
| PRSH_IND        | Indicates whether the item is perishable.                                                                 | VARCHAR2(1)     | 40          | No             |
| ITEM_TYPE_IDNT  | The unique identifier for the item type. Example item types include Swatch, Component, Raw, etc.          | VARCHAR2(6)     | 41          | No             |

| Name           | Description                                                                                                                                                                                                       | Data Type/Bytes | Field order | Required field |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| CONV_TYPE_IDNT | The unique identifier for the conveyable type.<br>Conveyable type indicates whether the product needs to be hand carried or can be placed on the conveyer belt to be moved.                                       | VARCHAR2(6)     | 42          | No             |
| CLLCTN_IDNT    | The unique identifier for the collection to which this item belongs.<br>A collection may be a line of leather furniture, including an armchair, ottoman, sofa, etc. which are all part of the Leather Collection. | VARCHAR2(6)     | 43          | No             |

**prditmldm.txt**

Business rules:

- This interface file contains one row for each item list. An item list is normally used to group items for reporting purpose.
- This interface file cannot contain duplicate records for an itemlst\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name          | Description                            | Data Type/Bytes | Field order | Required field |
|---------------|----------------------------------------|-----------------|-------------|----------------|
| ITEMLIST_IDNT | The unique identifier of an item list. | VARCHAR2(10)    | 1           | Yes            |



| Name          | Description                                           | Data Type/Bytes | Field order | Required field |
|---------------|-------------------------------------------------------|-----------------|-------------|----------------|
| CREATE_ID     | The login ID of the person who created the Item List. | VARCHAR2(30)    | 2           | Yes            |
| ITEMLIST_DESC | The description or name of the item list.             | VARCHAR2(120)   | 3           | No             |

**prditmlmdm.txt**

Business rules:

- This interface file contains the associations between item list and tracking level item identifiers.
- This interface file cannot contain duplicate records for an itemlst\_idnt and item\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name          | Description                            | Data Type/Bytes | Field order | Required field |
|---------------|----------------------------------------|-----------------|-------------|----------------|
| ITEMLIST_IDNT | The unique identifier of an item list. | VARCHAR2(10)    | 1           | Yes            |
| ITEM_IDNT     | The unique identifier of an item.      | VARCHAR2(25)    | 2           | Yes            |

**prditmltmdm.txt**

Business rules:

- This interface file contains associations among locations, tracking level items, and their location traits.
- This interface file cannot contain duplicate records for an item\_idnt, loc\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name      | Description                       | Data Type/Bytes | Field order | Required field |
|-----------|-----------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT | The unique identifier of an item. | VARCHAR2(25)    | 1           | Yes            |

| Name                 | Description                                                                                                  | Data Type/Bytes | Field order | Required field |
|----------------------|--------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_IDNT             | The unique identifier of the location.                                                                       | VARCHAR2(10)    | 2           | Yes            |
| LAUNCH_DT            | The date that the item should first be sold at the location.                                                 | DATE            | 3           | No             |
| DEPOSIT_CDE          | The code which indicates whether a deposit is associated with this item at the location                      | VARCHAR2(6)     | 4           | No             |
| FOOD_STAMP_IND       | Indicates whether the item is approved for food stamps at the location.                                      | VARCHAR2(1)     | 5           | No             |
| REWARD_ELIGIBLE_IND  | Indicates whether the item is legally valid for various types of bonus point/award programs at the location. | VARCHAR2(1)     | 6           | No             |
| NATL_BRAND_COMP_ITEM | The nationally branded item to which you would like to compare the current item.                             | VARCHAR2(25)    | 7           | No             |

| Name                  | Description                                                                                                  | Data Type/Bytes | Field order | Required field |
|-----------------------|--------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| STOP_SALE_IND         | Indicates that sale of the item should be stopped immediately at the location.                               | VARCHAR2(1)     | 8           | No             |
| ELECT_MKT_CLUBS       | The code that represents the electronic marketing clubs to which the item belongs at the location.           | VARCHAR2(6)     | 9           | No             |
| STORE_REORDERABLE_IND | Indicates whether the store may re-order the item.                                                           | VARCHAR2(1)     | 10          | No             |
| FULL_PALLET_ITEM_IND  | Indicates whether a store must reorder an item in full pallets only.                                         | VARCHAR2(1)     | 11          | No             |
| DEPOSIT_CDE_DESC      | The deposit code description which indicates whether a deposit is associated with this item at the location. | VARCHAR2(120)   | 12          | No             |

**prditmsmdm.txt**

Business rules:

- This interface file contains associations between a tracking level or above item, and a product season/phase.
- This interface file cannot contain duplicate records for an item.

- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name            | Description                                 | Data Type/Bytes | Field order | Required field |
|-----------------|---------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT       | The unique identifier of an item.           | VARCHAR2(25)    | 1           | Yes            |
| PROD_SEASN_IDNT | The unique identifier of a product season.  | VARCHAR2(3)     | 2           | Yes            |
| PROD_PHASE_IDNT | The unique identifier of the product phase. | VARCHAR2(3)     | 3           | Yes            |

### prditmuddm.txt

Business rules:

- This interface file contains the associations between user defined attributes (UDA) at the detail level.
- This interface file cannot contain duplicate records for an item\_uda\_dtl\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name               | Description                                                        | Data Type/Bytes | Field order | Required field |
|--------------------|--------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_UDA_HEAD_IDNT | The unique identifier of the UDA.                                  | CHARACTER(5)    | 1           | Yes            |
| ITEM_UDA_DTL_IDNT  | The unique identifier of the text or date or lov values for a uda. | VARCHAR2(256)   | 2           | Yes            |
| ITEM_UDA_DTL_DESC  | The description of UDA value, text, or date.                       | VARCHAR2(255)   | 3           | No             |

### prditmuhdm.txt

Business rules:

- This interface file contains distinct user defined attribute (UDA) values.
- This interface file cannot contain duplicate records for an item\_uda\_head\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name               | Description                                                                             | Data Type/Bytes | Field order | Required field |
|--------------------|-----------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_UDA_HEAD_IDNT | The unique identifier of the UDA.                                                       | VARCHAR2(5)     | 1           | Yes            |
| ITEM_UDA_TYPE_CDE  | The code designating the uda type:<br>DT=date, LV=list of values,<br>FF=Free form text. | VARCHAR2(3)     | 2           | Yes            |
| ITEM_UDA_HEAD_DESC | The description of the UDA.                                                             | VARCHAR2(120)   | 3           | Yes            |

**prditmumdm.txt**

Business rules:

- This interface file contains the associations between UDA (User Defined Attributes) at the detail level and item identifiers at the tracking level.
- This interface file cannot contain duplicate records for an item\_uda\_dtl\_idnt and item\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name               | Description                                                        | Data Type/Bytes | Field order | Required field |
|--------------------|--------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_UDA_HEAD_IDNT | The unique identifier of the UDA.                                  | CHARACTER(5)    | 1           | Yes            |
| ITEM_UDA_DTL_IDNT  | The unique identifier of the text or date or lov values for a uda. | CHARACTER(256)  | 2           | Yes            |
| ITEM_IDNT          | The unique identifier of an item.                                  | VARCHAR2(25)    | 3           | Yes            |

**prdpimdm.txt**

Business rules:

- This interface file contains the associations between packs and their component tracking-level item identifiers.

- This interface file cannot contain duplicate records for a pack\_idnt and item\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name          | Description                                    | Data Type/Bytes | Field order | Required field |
|---------------|------------------------------------------------|-----------------|-------------|----------------|
| PACK_IDNT     | The unique identifier of pack.                 | VARCHAR2(25)    | 1           | Yes            |
| PACK_ITEM_QTY | Total quantity of a unique item within a pack. | NUMBER(12,4)    | 2           | No             |
| ITEM_IDNT     | The unique identifier of an item.              | VARCHAR2(25)    | 3           | Yes            |

### **prdsbcdm.txt**

Business rules:

- This interface file contains a subclass within a class and a department.
- This interface file cannot contain duplicate records for a dept\_idnt, class\_idnt, subclass\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name         | Description                                                     | Data Type/Bytes | Field order | Required field |
|--------------|-----------------------------------------------------------------|-----------------|-------------|----------------|
| SBCLASS_IDNT | The unique identifier of the subclass in the product hierarchy. | VARCHAR2(4)     | 1           | Yes            |
| CLASS_IDNT   | The unique identifier of the class in the product hierarchy.    | VARCHAR2(4)     | 2           | Yes            |
| DEPT_IDNT    | The unique identifier of a department in the product hierarchy. | VARCHAR2(4)     | 3           | Yes            |
| SBCLASS_DESC | The name of the subclass in the product hierarchy.              | VARCHAR2(120)   | 4           | No             |

| Name              | Description                                                              | Data Type/Bytes | Field order | Required field |
|-------------------|--------------------------------------------------------------------------|-----------------|-------------|----------------|
| SBCLASS_BUYR_IDNT | The unique identifier of the buyer for this subclass of products.        | VARCHAR2(4)     | 5           | No             |
| SBCLASS_BUYR_NAME | The name of the buyer for this subclass of products.                     | VARCHAR2(120)   | 6           | No             |
| SBCLASS_MRCH_IDNT | The unique identifier for the merchandiser of this subclass of products. | VARCHAR2(4)     | 7           | No             |
| SBCLASS_MRCH_NAME | The name of the merchandiser for this subclass of products.              | VARCHAR2(120)   | 8           | No             |

### regngrpdm.txt

Business rules:

- This interface file contains regionality group information.
- This interface file cannot contain duplicate records for a regionality\_grp\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name                 | Description                                     | Data Type/Bytes | Field order | Required field |
|----------------------|-------------------------------------------------|-----------------|-------------|----------------|
| REGIONALITY_GRP_IDNT | The unique identifier of the regionality group. | VARCHAR2(4)     | 1           | Yes            |
| REGIONALITY_GRP_DESC | The name of the regionality group.              | VARCHAR2(120)   | 2           | No             |

| Name                      | Description                                                                                             | Data Type/Bytes | Field order | Required field |
|---------------------------|---------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| REGIONALITY_GRP_ROLE_CDE  | The role that a client wants to assign to this group. This field is referenced in the code type 'ROLE'. | VARCHAR2(6)     | 3           | No             |
| REGIONALITY_GRP_ROLE_DESC | The description for a role.                                                                             | VARCHAR2(120)   | 4           | No             |

### regnmtxdm.txt

Business rules:

- This interface file contains the associations among regionality groups, departments, locations and suppliers.
- This interface file cannot contain duplicate records for a regionality\_grp\_idnt, loc\_idnt, supp\_idnt, dept\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name                 | Description                                                           | Data Type/Bytes | Field order | Required field |
|----------------------|-----------------------------------------------------------------------|-----------------|-------------|----------------|
| REGIONALITY_GRP_IDNT | The unique identifier of the regionality group.                       | VARCHAR2(4)     | 1           | Yes            |
| LOC_IDNT             | The unique identifier of the location.                                | CHARACTER(10)   | 2           | Yes            |
| LOC_TYPE_CDE         | The code that indicates whether the location is a store or warehouse. | CHARACTER(2)    | 3           | Yes            |
| SUPP_IDNT            | The unique identifier of a supplier.                                  | CHARACTER(10)   | 4           | Yes            |



| Name      | Description                                                     | Data Type/Bytes | Field order | Required field |
|-----------|-----------------------------------------------------------------|-----------------|-------------|----------------|
| DEPT_IDNT | The unique identifier of a department in the product hierarchy. | CHARACTER(4)    | 5           | Yes            |

**rplcilddm.txt**

Business rules:

- If a dimension identifier is required but is not available, a value of -1 is needed.
- The banner\_idnt corresponding to the hdr\_media\_idnt and line\_media\_idnt must be the same.
- Cannot contain duplicate transactions for an item\_idnt, loc\_idnt, hdr\_media\_idnt, line\_media\_idnt, banner\_idnt, and day\_dt combination.
- Contains the replacement data for an item, location, order header media, and order line media combination on a given day.
- Follows the fact flat file interface layout standard.

| Name            | Description                                                                | Data Type/Bytes | Field Order | Required Field |
|-----------------|----------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT       | The unique identifier of an item.                                          | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT        | The unique identifier of the location.                                     | CHARACTER(10)   | 2           | Yes            |
| DAY_DT          | The transaction date when the customer order line was created or modified. | DATE            | 3           | Yes            |
| HDR_MEDIA_IDNT  | The unique identifier of the customer order header level media.            | CHARACTER(10)   | 4           | Yes            |
| LINE_MEDIA_IDNT | The unique identifier of the customer order line level media.              | CHARACTER(10)   | 5           | Yes            |
| BANNER_IDNT     | The unique identifier of a banner.                                         | CHARACTER(4)    | 6           | Yes            |

| Name                    | Description                                                                                                    | Data Type/Bytes | Field Order | Required Field |
|-------------------------|----------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_RPLC_IN_QTY           | The number of units that are received from the customer for a replacement in transaction.                      | NUMBER(12,4)    | 7           | No             |
| F_RPLC_OUT_QTY          | The number of units that are sent to the customer for a replacement out transaction.                           | NUMBER(12,4)    | 8           | No             |
| F_RPLC_COST_IN_AMT      | The total cost, in primary currency, of the units received from the customer for a replacement in transaction. | NUMBER(18,4)    | 9           | No             |
| F_RPLC_COST_IN_AMT_LCL  | The total cost, in local currency, of the units received from the customer for a replacement in transaction.   | NUMBER(18,4)    | 10          | No             |
| F_RPLC_COST_OUT_AMT     | The total cost, in primary currency, of the units sent to the customer for a replacement out transaction.      | NUMBER(18,4)    | 11          | No             |
| F_RPLC_COST_OUT_AMT_LCL | The total cost, in local currency, of the units sent to the customer for a replacement out transaction.        | NUMBER(18,4)    | 12          | No             |

### rsndm.txt

Business rules:

- This interface file contains the reason class, types, and codes for the reason dimension. The file can hold various kinds of transaction reasons/codes such as inventory adjustment, return-to-vendor, voids, sales, and so on. The reason class allows definition of the reason, and the corresponding types and codes can also be defined under the class.
- This interface file cannot contain duplicate records for a reasn\_code\_idnt, reasn\_type\_idnt, combination.

- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name             | Description                                | Data Type/Bytes | Field order | Required field |
|------------------|--------------------------------------------|-----------------|-------------|----------------|
| REASN_CODE_IDNT  | The unique identifier of the reason code.  | VARCHAR2(6)     | 1           | Yes            |
| REASN_TYPE_IDNT  | The unique identifier of the reason type.  | VARCHAR2(6)     | 2           | Yes            |
| REASN_CLASS_IDNT | The unique identifier of the reason class. | VARCHAR2(6)     | 3           | Yes            |
| REASN_CODE_DESC  | The description of the reason code         | VARCHAR2(120)   | 4           | No             |
| REASN_TYPE_DESC  | The description of the reason type.        | VARCHAR2(120)   | 5           | No             |
| REASN_CLASS_DESC | The description of the reason class        | VARCHAR2(120)   | 6           | No             |

### saviddm.txt

Business rules:

- This interface file contains summarized item availability quantities for a supplier, item on a given day.
- This interface file cannot contain duplicate transactions for an item\_idnt, supp\_idnt, and day\_dt combination.
- This interface file contains only the current day's new or changed information.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name      | Description                       | Data Type/Bytes | Field order | Required field |
|-----------|-----------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT | The unique identifier of an item. | CHARACTER(25)   | 1           | Yes            |

| Name        | Description                                                      | Data Type/Bytes | Field order | Required field |
|-------------|------------------------------------------------------------------|-----------------|-------------|----------------|
| SUPP_IDNT   | The unique identifier of a supplier.                             | CHARACTER(10)   | 2           | Yes            |
| DAY_DT      | The calendar day on which the transaction occurred.              | DATE            | 3           | Yes            |
| F_AVAIL_QTY | The quantity of stock available to be ordered from the supplier. | NUMBER(12,4)    | 4           | No             |

**scmialddm.txt**

Business rules:

- Contains data pertaining to a supplier's missed shipments by location and day.
- Cannot contain duplicate transactions for a supp\_idnt, loc\_idnt, day\_dt.
- Follows the fact flat file interface layout standard.

| Name               | Description                                                                                   | Data Type/Bytes | Field Order | Required Field |
|--------------------|-----------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| SUPP_IDNT          | The unique identifier of a supplier.                                                          | CHARACTER(10)   | 1           | Yes            |
| LOC_IDNT           | The unique identifier of the location.                                                        | CHARACTER(10)   | 2           | Yes            |
| DAY_DT             | The calendar day on which the transaction occurred.                                           | DATE            | 3           | Yes            |
| F_MISSED_ASN_COUNT | The total number of ASN (advanced ship notice) shipments that were expected and not received. | NUMBER(16,4)    | 4           | No             |

**scmidlddm.txt**

- Cannot contain duplicate transactions for a supp\_idnt, loc\_idnt, day\_dt.
- Contains data pertaining to a supplier's missed deliveries by location and day.
- Follows the fact flat file interface layout standard.

| Name                 | Description                                                    | Data Type/Bytes | Field Order | Required Field |
|----------------------|----------------------------------------------------------------|-----------------|-------------|----------------|
| SUPP_IDNT            | The unique identifier of a supplier.                           | CHARACTER(10)   | 1           | Yes            |
| LOC_IDNT             | The unique identifier of the location.                         | CHARACTER(10)   | 2           | Yes            |
| DAY_DT               | The calendar day on which the transaction occurred.            | DATE            | 3           | Yes            |
| F_MISSED_SCHED_COUNT | The total number of scheduled shipments that are not received. | NUMBER(16,4)    | 4           | No             |

**scmiolddm.txt**

Business rules:

- Cannot contain duplicate transactions for a supp\_idnt, loc\_idnt, day\_dt.
- Contains data pertaining to a supplier's missed purchase orders by location and day.
- Follows the fact flat file interface layout standard.

| Name                 | Description                                                                       | Data Type/Bytes | Field Order | Required Field |
|----------------------|-----------------------------------------------------------------------------------|-----------------|-------------|----------------|
| SUPP_IDNT            | The unique identifier of a supplier.                                              | CHARACTER(10)   | 1           | Yes            |
| LOC_IDNT             | The unique identifier of the location.                                            | CHARACTER(10)   | 2           | Yes            |
| DAY_DT               | The calendar day on which the transaction occurred.                               | DATE            | 3           | Yes            |
| F_MISSED_ORDER_COUNT | The total number of purchase order shipments that were expected and not received. | NUMBER(16,4)    | 4           | No             |

**scqcdm.txt**

Business rules:

- Cannot contain duplicate transactions for an item\_idnt, supp\_idnt, ship\_idnt, loc\_idnt, day\_dt, po\_idnt.
- Contains shipment information about which items requiring QC (quality control) failed or passed the QC test.

- Follows the fact flat file interface layout standard.

| Name            | Description                                                                    | Data Type/Bytes | Field Order | Required Field |
|-----------------|--------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT       | The unique identifier of an item.                                              | CHARACTER(25)   | 1           | Yes            |
| SHIP_IDNT       | The unique identifier of the shipment.                                         | CHARACTER(10)   | 2           | Yes            |
| SUPP_IDNT       | The unique identifier of a supplier.                                           | CHARACTER(10)   | 3           | Yes            |
| LOC_IDNT        | The unique identifier of the location.                                         | CHARACTER(10)   | 4           | Yes            |
| DAY_DT          | The calendar day on which the transaction occurred.                            | DATE            | 5           | Yes            |
| PO_IDNT         | The unique identifier of a purchase order.                                     | CHARACTER(8)    | 6           | Yes            |
| F_QC_FLAG       | Indicates whether or not quality control checking was required on the receipt. | CHARACTER(1)    | 7           | No             |
| F_QC_FAILED_QTY | The total quantity of items that failed quality control checks.                | NUMBER(12,4)    | 8           | No             |
| F_QC_PASSED_QTY | The total quantity of items that passed quality control checks.                | NUMBER(12,4)    | 9           | No             |

**scrtllddm.txt**

Business rules:

- This interface file contains shipment information about quantity of items received. This data is only associated with scrqllddm.txt.
- This interface file contains shipment information about timeliness of receipt. This data is only associated with scrtllddm.txt.
- This interface file contains shipment information about which items requiring QC (quality control) failed or passed the QC test. This data is only associated with scqcdm.txt.

- This interface file cannot contain duplicate transactions for item\_idnt, ship\_idnt, supp\_idnt, loc\_idnt, day\_dt, po\_idnt. This interface file is also applied to the scrqtllddm.txt and scrtrllddm.txt interface files.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name            | Description                                                                                                                                    | Data Type/Bytes | Field Order | Required Field |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT       | The unique identifier of an item.                                                                                                              | CHARACTER(25)   | 1           | Yes            |
| SUPP_IDNT       | The unique identifier of a supplier.                                                                                                           | CHARACTER(10)   | 2           | Yes            |
| SHIP_IDNT       | The unique identifier of the shipment.                                                                                                         | CHARACTER(10)   | 3           | Yes            |
| LOC_IDNT        | The unique identifier of the location.                                                                                                         | CHARACTER(10)   | 4           | Yes            |
| DAY_DT          | The calendar day on which the transaction occurred.                                                                                            | DATE            | 5           | Yes            |
| PO_IDNT         | The unique identifier of a purchase order.                                                                                                     | CHARACTER(8)    | 6           | Yes            |
| F_ON_TIME_COUNT | The number of deliveries where the quantity received equaled the number expected. In this day-level table, the count value can only be 0 or 1. | NUMBER(16,4)    | 7           | No             |
| F_EARLY_COUNT   | The number of deliveries that arrived before the scheduled time. In this day-level table, the count value can only be 0 or 1.                  | NUMBER(16,4)    | 8           | No             |

| Name               | Description                                                                                                                               | Data Type/Bytes | Field Order | Required Field |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_LATE_COUNT       | The number of deliveries that arrived after the scheduled time. In this day-level table, the count value can only be 0 or 1.              | NUMBER(16,4)    | 9           | No             |
| F_UNSCHEd_COUNT    | The number of deliveries that arrived on days other than the scheduled date. In this day-level table, the count value can only be 0 or 1. | NUMBER(16,4)    | 10          | No             |
| F_DAYS_EARLY_COUNT | The total number of days a shipment arrived before the scheduled date.                                                                    | NUMBER(16,4)    | 11          | No             |
| F_DAYS_LATE_COUNT  | The total number of days a shipment arrived after the scheduled date.                                                                     | NUMBER(16,4)    | 12          | No             |

### scrqtlldm.txt

Business rules:

- Contains shipment information about quantity of items received.
- Cannot contain duplicate transactions for an item\_idnt, supp\_idnt, ship\_idnt, loc\_idnt, day\_dt, po\_idnt.
- Follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name      | Description                            | Data Type/Bytes | Field Order | Required Field |
|-----------|----------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT | The unique identifier of an item.      | CHARACTER(25)   | 1           | Yes            |
| SUPP_IDNT | The unique identifier of a supplier.   | CHARACTER(10)   | 2           | Yes            |
| SHIP_IDNT | The unique identifier of the shipment. | CHARACTER(10)   | 3           | Yes            |



| Name                 | Description                                                                                                                                                                          | Data Type/Bytes | Field Order | Required Field |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| LOC_IDNT             | The unique identifier of the location.                                                                                                                                               | CHARACTER(10)   | 4           | Yes            |
| DAY_DT               | The calendar day on which the transaction occurred.                                                                                                                                  | DATE            | 5           | Yes            |
| PO_IDNT              | The unique identifier of a purchase order.                                                                                                                                           | CHARACTER(8)    | 6           | Yes            |
| F_ASN_EXPECTED_QTY   | The total advanced shipment notice (ASN) quantity expected.                                                                                                                          | NUMBER(12,4)    | 7           | No             |
| F_RECEIVED_QTY       | The total quantity received.                                                                                                                                                         | NUMBER(12,4)    | 8           | No             |
| F_ORDERED_QTY        | The total quantity ordered.                                                                                                                                                          | NUMBER(12,4)    | 9           | No             |
| F_ASN_EXPECTED_COUNT | The number of advance shipping notice (ASN) deliveries where the quantity received equaled the quantity expected. The count value can only be 0 or 1.                                | NUMBER(16,4)    | 10          | No             |
| F_ASN_UNDER_COUNT    | The number of advanced shipping notice (ASN) deliveries where the quantity received were less than the number expected. In this day-level table, the count value can only be 0 or 1. | NUMBER(16,4)    | 11          | No             |

| Name               | Description                                                                                                                                                                    | Data Type/Bytes | Field Order | Required Field |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_ASN_OVER_COUNT   | The number of advanced shipping notice (ASN) deliveries where the quantity received exceeded the number expected. In this day-level table, the count value can only be 0 or 1. | NUMBER(16,4)    | 12          | No             |
| F_MISMATCHED_COUNT | The number of deliveries where quantity was received for an item that was not expected. In this day-level table, the count value can only be 0 or 1.                           | NUMBER(16,4)    | 13          | No             |
| F_FULL_PO_COUNT    | The number of purchase orders where all expected quantity was received. In this day-level table, the count value can only be 0 or 1.                                           | NUMBER(16,4)    | 14          | No             |
| F_PART_PO_COUNT    | The number of purchase orders where only part of the expected quantity was received. In this day-level table, the count value can only be 0 or 1.                              | NUMBER(16,4)    | 15          | No             |
| F_OVER_PO_COUNT    | The number of purchase orders where more than the expected quantity was received. In this day-level table, the count value can only be 0 or 1.                                 | NUMBER(16,4)    | 16          | No             |

| Name       | Description                                                  | Data Type/Bytes | Field Order | Required Field |
|------------|--------------------------------------------------------------|-----------------|-------------|----------------|
| PICKUP_LOC | The user-entered location of shipment for client to pick up. | CHARACTER(250)  | 17          | No             |
| PICKUP_NBR | The user-entered identifier of a shipment.                   | CHARACTER(25)   | 18          | No             |
| PICKUP_DT  | The user entered date of the pickup.                         | DATE            | 19          | No             |

**sctidm.txt**

Business rules:

- This interface file contains supplier contract information.
- This interface file cannot contain duplicate transactions for an item\_idnt, cntrect\_idnt, day\_dt combination.
- This interface file contains only the current day's new or changed information.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name              | Description                                                    | Data Type/Bytes | Field order | Required field |
|-------------------|----------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT         | The unique identifier of an item.                              | CHARACTER(25)   | 1           | Yes            |
| CNTRCT_IDNT       | The unique identifier of a contract.                           | CHARACTER(6)    | 2           | Yes            |
| DAY_DT            | The calendar day on which the transaction occurred.            | DATE            | 3           | Yes            |
| F_CNTRCT_QTY      | The total contracted quantity to be ordered from the supplier. | NUMBER(12,4)    | 4           | No             |
| F_CNTRCT_COST_AMT | The unit purchase cost negotiated for this contract.           | NUMBER(18,4)    | 5           | No             |

| Name                         | Description                                                                                                | Data Type/Bytes | Field order | Required field |
|------------------------------|------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_CNTRCT_ORD_QTY             | The total ordered quantity from the contract to date for all locations.                                    | NUMBER(12,4)    | 6           | No             |
| F_CNTRCT_ORD_COST_AMT        | The total cost value for the ordered quantity from the contract to date for all locations.                 | NUMBER(18,4)    | 7           | No             |
| F_CNTRCT_ORD_CNCLLD_QTY      | The total cancelled quantities from the contract to date, for all locations and orders.                    | NUMBER(12,4)    | 8           | No             |
| F_CNTRCT_ORD_CNCLLD_COST_AMT | The total cost value for the cancelled quantities from the contract to date, for all locations and orders. | NUMBER(18,4)    | 9           | No             |

### **seasndm.txt**

Business rules:

- This interface file contains seasons. Seasons are arbitrary periods of time around which some retailers organize their buying and selling patterns. Each day should fall within no more than one season.
- This interface file cannot contain duplicate records for a seasn\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name       | Description                        | Data Type/Bytes | Field order | Required field |
|------------|------------------------------------|-----------------|-------------|----------------|
| SEASN_IDNT | The unique identifier of a season. | VARCHAR2(3)     | 1           | Yes            |

| Name           | Description                             | Data Type/Bytes | Field order | Required field |
|----------------|-----------------------------------------|-----------------|-------------|----------------|
| SEASN_START_DT | The beginning date for the season.      | DATE            | 2           | Yes            |
| SEASN_END_DT   | The ending date for the season.         | DATE            | 3           | Yes            |
| SEASN_DESC     | The description or name for the season. | VARCHAR2(120)   | 4           | No             |

**sfclwdm.txt**

Business rules:

- This interface file contains sales forecast information for an item and location combination on a given week.
- This interface file cannot contain duplicate transactions for an item\_idnt, loc\_idnt, and day\_dt.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name           | Description                                         | Data Type/Bytes | Field order | Required field |
|----------------|-----------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT      | The unique identifier of an item.                   | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT       | The unique identifier of the location.              | CHARACTER(10)   | 2           | Yes            |
| DAY_DT         | The calendar day on which the transaction occurred. | DATE            | 3           | Yes            |
| F_FCST_SLS_QTY | The forecast sales quantity.                        | NUMBER(12,4)    | 4           | No             |

**silsildmdm.txt**

Business rules:

- This interface file contains sales and returns for an item, location, day, minute, voucher, and transaction.
- Assumes that tran\_idnts received from the source system are unique across media-location-register-employee-minute-day. In an example from brick and mortar, two items, sold at the same location, by the same employee in the same minute, but at two different cash registers to two different customers in two different transactions, stores result in two separate and distinct tran\_idnts; similarly, the same item/loc/day/minute/register but different employees, ringing up two separate transactions stores result in two distinct tran\_idnts.
- tran\_idnt is unique across all locations.
- The format of the min\_idnt field is the hour (in format HH24) followed by a number 01-60, which indicates the minute of that hour.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.

| Name      | Description                                                                                                                                                                                                  | Data Type/Bytes | Field order | Required field |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT | The unique identifier of an item.                                                                                                                                                                            | CHARACTER(25)   | 1           | Yes            |
| TRAN_IDNT | The unique identifier of the transaction.                                                                                                                                                                    | VARCHAR2(30)    | 2           | Yes            |
| VCHR_IDNT | Voucher number. If the Item is a gift certificate, then the corresponding Item Number stores represent a VCHR_IDNT. This attribute is not a dimensional attribute but is used to uniquely identify a record. | CHARACTER(25)   | 3           | Yes            |
| DAY_DT    | The calendar day on which the transaction occurred.                                                                                                                                                          | DATE            | 4           | Yes            |
| MIN_IDNT  | The unique identifier of the minute.                                                                                                                                                                         | NUMBER(4)       | 5           | Yes            |

| Name                     | Description                                                                                                                                | Data Type/Bytes | Field order | Required field |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| OVERRIDE_REASN_CODE_IDNT | The unique identifier for a reason code.                                                                                                   | CHARACTER(6)    | 6           | Yes            |
| OVERRIDE_REASN_TYPE_IDNT | The unique identifier for a reason type.                                                                                                   | CHARACTER(6)    | 7           | Yes            |
| LOC_IDNT                 | The unique identifier of the location.                                                                                                     | CHARACTER(10)   | 8           | Yes            |
| RTRN_REASN_IDNT          | The unique identifier used to identify a return reason code. These codes should exist in the RMS CODE_DETAIL table under 'SARR' code type. | CHARACTER(6)    | 9           | Yes            |
| CUST_REF                 | The customer identifier associated with the transaction.                                                                                   | CHARACTER(20)   | 10          | Yes            |
| CUST_REF_TYPE            | The type of the identifier number used by a customer.                                                                                      | CHARACTER(6)    | 11          | Yes            |
| EMPLY_IDNT               | The unique identifier of the employee.                                                                                                     | CHARACTER(10)   | 12          | Yes            |
| SLSPRSN_IDNT             | The unique identifier for a salesperson.                                                                                                   | CHARACTER(10)   | 13          | Yes            |
| CSHR_IDNT                | The unique identifier for a cashier.                                                                                                       | CHARACTER(10)   | 14          | Yes            |
| RGSTR_IDNT               | The unique identifier of the register.                                                                                                     | CHARACTER(10)   | 15          | Yes            |

| <b>Name</b>        | <b>Description</b>                                                       | <b>Data Type/Bytes</b> | <b>Field order</b> | <b>Required field</b> |
|--------------------|--------------------------------------------------------------------------|------------------------|--------------------|-----------------------|
| REASN_CODE_IDNT    | The unique identifier of the reason code.                                | CHARACTER(6)           | 16                 | Yes                   |
| REASN_TYPE_IDNT    | The unique identifier of the reason type.                                | CHARACTER(6)           | 17                 | Yes                   |
| SUB_TRAN_TYPE_IDNT | The unique identifier of the sub-transaction type.                       | CHARACTER(6)           | 18                 | Yes                   |
| LINE_MEDIA_IDNT    | The identifier of a customer order line media.                           | CHARACTER(10)          | 19                 | Yes                   |
| BANNER_IDNT        | The unique identifier of a banner.                                       | CHARACTER(4)           | 20                 | Yes                   |
| SELLING_ITEM_IDNT  | The unique identifier of a selling item.                                 | CHARACTER(25)          | 21                 | Yes                   |
| CO_HDR_IDNT        | The unique identifier of a customer order.                               | CHARACTER(30)          | 22                 | Yes                   |
| CO_LINE_IDNT       | The unique identifier of a customer order line.                          | CHARACTER(30)          | 23                 | Yes                   |
| DROP_SHIP_IND      | An indicator to identify if an item is shipped directly to the customer. | CHARACTER(1)           | 24                 | No                    |
| RTL_TYPE_CDE       | The price type ('R'egular, 'P'romotion, 'C'learance).                    | CHARACTER(2)           | 25                 | Yes                   |
| F_SLS_AMT          | The value of the sale in primary currency                                | NUMBER(18,4)           | 26                 | No                    |
| F_SLS_AMT_LCL      | The value of the sale in local currency                                  | NUMBER(18,4)           | 27                 | No                    |



| Name                   | Description                                                          | Data Type/Bytes | Field order | Required field |
|------------------------|----------------------------------------------------------------------|-----------------|-------------|----------------|
| F_SLS_QTY              | The number of items involved in the sale                             | NUMBER(12,4)    | 28          | No             |
| F_SLS_PRFT_AMT         | The profit amount realized on the sale in primary currency.          | NUMBER(18,4)    | 29          | No             |
| F_SLS_PRFT_AMT_LCL     | The profit amount realized on the sale in local currency.            | NUMBER(18,4)    | 30          | No             |
| F_RTRN_AMT             | The value of the return in primary currency                          | NUMBER(18,4)    | 31          | No             |
| F_RTRN_AMT_LCL         | The value of the return in local currency                            | NUMBER(18,4)    | 32          | No             |
| F_RTRN_QTY             | The number of items involved in the return                           | NUMBER(12,4)    | 33          | No             |
| F_RTRN_PRFT_AMT        | The profit amount realized on the return in primary currency         | NUMBER(18,4)    | 34          | No             |
| F_RTRN_PRFT_AMT_LCL    | The profit amount realized on the return in local currency           | NUMBER(18,4)    | 35          | No             |
| F_SLS_ENTER_ITEM_COUNT | The number of times the item is manually entered by cashier for sale | NUMBER(16,4)    | 36          | No             |
| F_SLS_SCAN_ITEM_COUNT  | The number of times the item is scanned by cashier for sale          | NUMBER(16,4)    | 37          | No             |

| Name                    | Description                                                                  | Data Type/Bytes | Field order | Required field |
|-------------------------|------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_RTRN_ENTER_ITEM_COUNT | The number of times the item is manually entered by cashier for return       | NUMBER(16,4)    | 38          | No             |
| F_RTRN_SCAN_ITEM_COUNT  | Number of times the item is scanned by cashier for return                    | NUMBER(16,4)    | 39          | No             |
| F_SLS_IS_MKUP_COUNT     | The count of the number of in store markup sales transactions                | NUMBER(16,4)    | 40          | No             |
| F_SLS_IS_MKDN_COUNT     | The count of the number of in store markdown sales transactions              | NUMBER(16,4)    | 41          | No             |
| F_RTRN_IS_MKUP_COUNT    | The count of the number of in store markup return transactions               | NUMBER(16,4)    | 42          | No             |
| F_RTRN_IS_MKDN_COUNT    | The count of the number of in store markdown return transactions             | NUMBER(16,4)    | 43          | No             |
| F_SLS_IS_MKUP_AMT       | The total in store markup amount in primary currency for sales transactions  | NUMBER(18,4)    | 44          | No             |
| F_SLS_IS_MKUP_AMT_LCL   | The total in store markup amount in local currency for sales transactions    | NUMBER(18,4)    | 45          | No             |
| F_RTRN_IS_MKUP_AMT      | The total in store markup amount in primary currency for return transactions | NUMBER(18,4)    | 46          | No             |

| Name                     | Description                                                                          | Data Type/Bytes | Field order | Required field |
|--------------------------|--------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_RTRN_IS_MKUP_AMT_LCL   | The total in store markup amount in local currency for return transactions           | NUMBER(18,4)    | 47          | No             |
| F_SLS_IS_MKDN_AMT        | The total in store markdown amount in primary currency for sales transactions        | NUMBER(18,4)    | 48          | No             |
| F_SLS_IS_MKDN_AMT_LCL    | The total in store markdown amount in local currency for sales transactions          | NUMBER(18,4)    | 49          | No             |
| F_RTRN_IS_MKDN_AMT       | The total in store markdown amount in primary currency for return transactions       | NUMBER(18,4)    | 50          | No             |
| F_RTRN_IS_MKDN_AMT_LCL   | The total in store markdown amount in local currency for return transactions         | NUMBER(18,4)    | 51          | No             |
| F_SLS_EMPTY_DISC_AMT     | The total employee retail discount amount in primary currency for sales transactions | NUMBER(18,4)    | 52          | No             |
| F_SLS_EMPTY_DISC_AMT_LCL | The total employee retail discount amount in local currency for sales transactions   | NUMBER(18,4)    | 53          | No             |

| Name                      | Description                                                                                                 | Data Type/Bytes | Field order | Required field |
|---------------------------|-------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_RTRN_EMPTY_DISC_AMT     | The total employee retail discount amount in primary currency for return transactions                       | NUMBER(18,4)    | 54          | No             |
| F_RTRN_EMPTY_DISC_AMT_LCL | The total employee retail discount amount in local currency for return transactions                         | NUMBER(18,4)    | 55          | No             |
| F_SLS_ACCOM_AMT           | The total customer order accommodations, associated with items, in primary currency for sales transactions. | NUMBER(18,4)    | 56          | No             |
| F_SLS_ACCOM_AMT_LCL       | The total customer order accommodations, associated with items, in local currency for sales transactions.   | NUMBER(18,4)    | 57          | No             |
| F_SLS_VAT_AMT             | The value of the sales value added tax in primary currency.                                                 | NUMBER(18,4)    | 58          | No             |
| F_SLS_VAT_AMT_LCL         | The value of the sales value added tax in local currency                                                    | NUMBER(18,4)    | 59          | No             |
| F_RTRN_VAT_AMT            | The value of the return value added tax in primary currency                                                 | NUMBER(18,4)    | 60          | No             |

| Name               | Description                                               | Data Type/Bytes | Field order | Required field |
|--------------------|-----------------------------------------------------------|-----------------|-------------|----------------|
| F_RTRN_VAT_AMT_LCL | The value of the return value added tax in local currency | NUMBER(18,4)    | 61          | No             |

**sismkdnildm.txt**

Business rules:

- This interface file contains point of sale, permanent, and clearance markdown and markup information for an item, location, and retail type on a given day.
- This interface file cannot contain duplicate transactions for a item\_idnt, loc\_idnt, rtl\_type\_cde, day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- This interface file contains neither break-to-sell items nor packs that contain break-to-sell component items.
- Typical markdowns, markups, markdown cancels, and markup cancels should be positive values in their respective fields. Any reversals of the transactions that use the same tran data codes contain negative values in those applicable fields.

| Name           | Description                                           | Data Type/Bytes | Field order | Required field |
|----------------|-------------------------------------------------------|-----------------|-------------|----------------|
| ITEM_IDNT      | The unique identifier of an item.                     | CHARACTER(25)   | 1           | Yes            |
| LOC_IDNT       | The unique identifier of the location.                | CHARACTER(10)   | 2           | Yes            |
| RTL_TYPE_CDE   | The price type ('R'egular, 'P'romotion, 'C'learance). | CHARACTER(2)    | 3           | Yes            |
| DAY_DT         | The calendar day on which the transaction occurred.   | DATE            | 4           | Yes            |
| F_MKDN_AMT     | The value of the markdown, in primary currency.       | NUMBER(18,4)    | 5           | No             |
| F_MKDN_AMT_LCL | The value of the markdown, in local currency.         | NUMBER(18,4)    | 6           | No             |

| <b>Name</b>         | <b>Description</b>                                     | <b>Data Type/Bytes</b> | <b>Field order</b> | <b>Required field</b> |
|---------------------|--------------------------------------------------------|------------------------|--------------------|-----------------------|
| F_MKDN_QTY          | The quantity of the markdown                           | NUMBER(12,4)           | 7                  | No                    |
| F_MKUP_AMT          | The value of the markup, in primary currency.          | NUMBER(18,4)           | 8                  | No                    |
| F_MKUP_AMT_LCL      | The value of the markup, in local currency.            | NUMBER(18,4)           | 9                  | No                    |
| F_MKUP_QTY          | The quantity of the markup.                            | NUMBER(12,4)           | 10                 | No                    |
| F_MKDN_CNCL_AMT     | The value of the markdown cancel, in primary currency. | NUMBER(18,4)           | 11                 | No                    |
| F_MKDN_CNCL_AMT_LCL | The value of the markdown cancel, in local currency.   | NUMBER(18,4)           | 12                 | No                    |
| F_MKDN_CNCL_QTY     | The quantity of the markdown cancel.                   | NUMBER(12,4)           | 13                 | No                    |
| F_MKUP_CNCL_AMT     | The value of the markup cancel, in primary currency.   | NUMBER(18,4)           | 14                 | No                    |
| F_MKUP_CNCL_AMT_LCL | The value of the markup cancel, in local currency.     | NUMBER(18,4)           | 15                 | No                    |
| F_MKUP_CNCL_QTY     | The quantity of the markup cancel.                     | NUMBER(12,4)           | 16                 | No                    |

### **slsprmilmdm.txt**

Business rules:

- If a dimension identifier is required but is not available, a value of -1 is needed.
- TRAN\_IDNT is unique across all locations.
- Follows the fact flat file interface layout standard.

| Name              | Description                                                                                   | Data Type/Bytes | Field order | Required field |
|-------------------|-----------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| TRAN_IDNT         | The unique identifier of a sales transaction.                                                 | VARCHAR2(30)    | 1           | Yes            |
| ITEM_IDNT         | The unique identifier of an item.                                                             | CHARACTER(25)   | 2           | Yes            |
| DAY_DT            | The calendar day on which the transaction occurred.                                           | DATE            | 3           | Yes            |
| MIN_IDNT          | The unique identifier of the minute. This is the minute the sales transaction was created.    | NUMBER(4)       | 4           | Yes            |
| PRMTN_DTL_IDNT    | The identifier of the promotion detail.                                                       | CHARACTER(10)   | 5           | Yes            |
| HEAD_IDNT         | The unique identifier of the promotion.                                                       | CHARACTER(10)   | 6           | Yes            |
| PRMTN_SRC_CDE     | The unique identifier of the promotion source. The valid value can be 'DTC', 'RMS' or others. | CHARACTER(6)    | 7           | Yes            |
| SELLING_ITEM_IDNT | The unique identifier of a selling item.                                                      | CHARACTER(25)   | 8           | Yes            |

| <b>Name</b>          | <b>Description</b>                                            | <b>Data Type/Bytes</b> | <b>Field order</b> | <b>Required field</b> |
|----------------------|---------------------------------------------------------------|------------------------|--------------------|-----------------------|
| LINE_MEDIA_IDNT      | The unique identifier of the customer order line level media. | CHARACTER(10)          | 9                  | Yes                   |
| BANNER_IDNT          | The unique identifier of a banner.                            | CHARACTER(4)           | 10                 | Yes                   |
| LOC_IDNT             | The unique identifier of the location.                        | CHARACTER(10)          | 11                 | Yes                   |
| CUST_REF             | The customer identifier associated with the transaction.      | CHARACTER(20)          | 12                 | Yes                   |
| CUST_REF_TYPE        | The type of the identifier number used by a customer.         | CHARACTER(6)           | 13                 | Yes                   |
| CO_LINE_IDNT         | The unique identifier of a customer order line.               | VARCHAR2(30)           | 14                 | Yes                   |
| CO_HDR_IDNT          | The unique identifier of a customer order header.             | VARCHAR2(30)           | 15                 | Yes                   |
| F_PRMTN_MKDN_AMT     | The promotional markdown amount in primary currency.          | NUMBER(18,4)           | 16                 | No                    |
| F_PRMTN_MKDN_AMT_LCL | The promotional markdown amount in local currency.            | NUMBER(18,4)           | 17                 | No                    |



**stlblmthdm.txt**

Business rules:

- This interface file contains stock ledger values for a department, class, subclass, and location on a given month.
- This interface file cannot contain duplicate transactions for a dept\_idnt, class\_idnt, subclass\_idnt, loc\_idnt, and day\_dt combination.
- This interface file can only be populated for one time, either Gregorian time or 454 time.
- This interface file follows the fact flat file interface layout standard.

| Name                   | Description                                                           | Data Type/Bytes | Field order | Required field |
|------------------------|-----------------------------------------------------------------------|-----------------|-------------|----------------|
| SBCLASS_IDNT           | The unique identifier of the subclass in the product hierarchy.       | CHARACTER(4)    | 1           | Yes            |
| CLASS_IDNT             | The unique identifier of the class in the product hierarchy.          | CHARACTER(4)    | 2           | Yes            |
| DEPT_IDNT              | The unique identifier of a department in the product hierarchy.       | CHARACTER(4)    | 3           | Yes            |
| LOC_IDNT               | The unique identifier of the location.                                | CHARACTER(10)   | 4           | Yes            |
| LOC_TYPE_CDE           | The code that indicates whether the location is a store or warehouse. | CHARACTER(2)    | 5           | Yes            |
| DAY_DT                 | The calendar day on which the transaction occurred.                   | DATE            | 6           | Yes            |
| F_IVL_BEG_SOH_COST_AMT | The beginning of period stock on hand total cost, in primary currency | NUMBER(18,4)    | 7           | N              |

| Name                       | Description                                                             | Data Type/Bytes | Field order | Required field |
|----------------------------|-------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_BEG_SOH_COST_AMT_LCL | The beginning of period stock on hand total cost, in local currency     | NUMBER(18,4)    | 8           | N              |
| F_IVL_BEG_SOH_RTL_AMT      | The beginning of period stock on hand total retail, in primary currency | NUMBER(18,4)    | 9           | N              |
| F_IVL_BEG_SOH_RTL_AMT_LCL  | The beginning of period stock on hand total retail, in local currency   | NUMBER(18,4)    | 10          | N              |
| F_IVL_SOH_ADJ_COST_AMT     | The value at cost of stock on hand adjustments, in primary currency.    | NUMBER(18,4)    | 11          | N              |
| F_IVL_SOH_ADJ_COST_AMT_LCL | The value at cost of stock on hand adjustments, in local currency.      | NUMBER(18,4)    | 12          | N              |
| F_IVL_SOH_ADJ_RTL_AMT      | The value at retail of stock on hand adjustments, in primary currency   | NUMBER(18,4)    | 13          | N              |
| F_IVL_SOH_ADJ_RTL_AMT_LCL  | The value at retail of stock on hand adjustments, in local currency     | NUMBER(18,4)    | 14          | N              |
| F_IVL_RCPTS_COST_AMT       | The value at cost of inventory received, in primary currency            | NUMBER(18,4)    | 15          | N              |
| F_IVL_RCPTS_COST_AMT_LCL   | The value at cost of inventory received, in local currency              | NUMBER(18,4)    | 16          | N              |

| Name                      | Description                                                                 | Data Type/Bytes | Field order | Required field |
|---------------------------|-----------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_RCPTS_RTL_AMT       | The value at retail of inventory received, in primary currency              | NUMBER(18,4)    | 17          | N              |
| F_IVL_RCPTS_RTL_AMT_LCL   | The value at retail of inventory received, in local currency                | NUMBER(18,4)    | 18          | N              |
| F_IVL_RTV_COST_AMT        | The value at cost of inventory returned to a vendor, in primary currency    | NUMBER(18,4)    | 19          | N              |
| F_IVL_RTV_COST_AMT_LCL    | The value at cost of inventory returned to a vendor, in local currency.     | NUMBER(18,4)    | 20          | N              |
| F_IVL_RTV_RTL_AMT         | The value at retail of inventory returned to a vendor, in primary currency. | NUMBER(18,4)    | 21          | N              |
| F_IVL_RTV_RTL_AMT_LCL     | The value at retail of inventory returned to a vendor, in local currency.   | NUMBER(18,4)    | 22          | N              |
| F_IVL_TSF_IN_COST_AMT     | The value at cost of inventory transferred in, in primary currency          | NUMBER(18,4)    | 23          | N              |
| F_IVL_TSF_IN_COST_AMT_LCL | The value at cost of inventory transferred in, in local currency            | NUMBER(18,4)    | 24          | N              |

| Name                       | Description                                                                                   | Data Type/Bytes | Field order | Required field |
|----------------------------|-----------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_TSF_IN_RTL_AMT       | The value at retail of inventory transferred in, in primary currency                          | NUMBER(18,4)    | 25          | N              |
| F_IVL_TSF_IN_RTL_AMT_LCL   | The value at retail of inventory transferred in, in local currency.                           | NUMBER(18,4)    | 26          | N              |
| F_IVL_TSF_OUT_COST_AMT     | The value at cost of inventory transferred out, in primary currency                           | NUMBER(18,4)    | 27          | N              |
| F_IVL_TSF_OUT_COST_AMT_LCL | The value at cost of inventory transferred out, in local currency                             | NUMBER(18,4)    | 28          | N              |
| F_IVL_TSF_OUT_RTL_AMT      | The value at retail of inventory transferred out, in primary currency                         | NUMBER(18,4)    | 29          | N              |
| F_IVL_TSF_OUT_RTL_AMT_LCL  | The value at retail of inventory transferred out, in local currency                           | NUMBER(18,4)    | 30          | N              |
| F_IVL_SHRK_COST_AMT        | The value at cost of the difference between actual and ending inventory, in primary currency. | NUMBER(18,4)    | 31          | N              |
| F_IVL_SHRK_COST_AMT_LCL    | The value at cost of the difference between actual and ending inventory, in local currency.   | NUMBER(18,4)    | 32          | N              |

| Name                      | Description                                                                                     | Data Type/Bytes | Field order | Required field |
|---------------------------|-------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_SHRK_RTL_AMT        | The value at retail of the difference between actual and ending inventory, in primary currency. | NUMBER(18,4)    | 33          | N              |
| F_IVL_SHRK_RTL_AMT_LCL    | The value at retail of the difference between actual and ending inventory, in local currency.   | NUMBER(18,4)    | 34          | N              |
| F_IVL_RTRNS_COST_AMT      | The value at cost of inventory returned from sales, in primary currency                         | NUMBER(18,4)    | 35          | N              |
| F_IVL_RTRNS_COST_AMT_LCL  | The value at cost of inventory returned from sales, in local currency                           | NUMBER(18,4)    | 36          | N              |
| F_IVL_RTRNS_RTL_AMT       | The value at retail of inventory returned from sales, in primary currency                       | NUMBER(18,4)    | 37          | N              |
| F_IVL_RTRNS_RTL_AMT_LCL   | The value at retail of inventory returned from sales, in local currency                         | NUMBER(18,4)    | 38          | N              |
| F_IVL_RECLASS_IN_COST_AMT | The value at cost of inventory reclassified to this location, in primary currency               | NUMBER(18,4)    | 39          | N              |

| Name                           | Description                                                                           | Data Type/Bytes | Field order | Required field |
|--------------------------------|---------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_RECLASS_IN_COST_AMT_LCL  | The value at cost of inventory reclassified to this location, in local currency       | NUMBER(18,4)    | 40          | N              |
| F_IVL_RECLASS_IN_RTL_AMT       | The value at retail of inventory reclassified to this location, in primary currency   | NUMBER(18,4)    | 41          | N              |
| F_IVL_RECLASS_IN_RTL_AMT_LCL   | The value at retail of inventory reclassified to this location, in local currency     | NUMBER(18,4)    | 42          | N              |
| F_IVL_RECLASS_OUT_COST_AMT     | The value at cost of inventory reclassified from this location, in primary currency   | NUMBER(18,4)    | 43          | N              |
| F_IVL_RECLASS_OUT_COST_AMT_LCL | The value at cost of inventory reclassified from this location, in local currency     | NUMBER(18,4)    | 44          | N              |
| F_IVL_RECLASS_OUT_RTL_AMT      | The value at retail of inventory reclassified from this location, in primary currency | NUMBER(18,4)    | 45          | N              |
| F_IVL_RECLASS_OUT_RTL_AMT_LCL  | The value at retail of inventory reclassified from this location, in local currency   | NUMBER(18,4)    | 46          | N              |
| F_IVL_SLS_COST_AMT             | The value at cost of inventory sold, in primary currency                              | NUMBER(18,4)    | 47          | N              |

| Name                       | Description                                                        | Data Type/Bytes | Field order | Required field |
|----------------------------|--------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_SLS_COST_AMT_LCL     | The value at cost of inventory sold, in local currency.            | NUMBER(18,4)    | 48          | N              |
| F_IVL_SLS_RTL_AMT          | The value at retail of inventory sold, in primary currency         | NUMBER(18,4)    | 49          | N              |
| F_IVL_SLS_RTL_AMT_LCL      | The value at retail of inventory sold, in local currency.          | NUMBER(18,4)    | 50          | N              |
| F_IVL_END_SOH_COST_AMT     | The end of period stock on hand total cost, in primary currency.   | NUMBER(18,4)    | 51          | N              |
| F_IVL_END_SOH_COST_AMT_LCL | The end of period stock on hand total cost, in local currency      | NUMBER(18,4)    | 52          | N              |
| F_IVL_END_SOH_RTL_AMT      | The end of period stock on hand total retail, in primary currency. | NUMBER(18,4)    | 53          | N              |
| F_IVL_END_SOH_RTL_AMT_LCL  | The end of period stock on hand total retail, in local currency    | NUMBER(18,4)    | 54          | N              |
| F_IVL_GRS_PRFT_AMT         | The total gross profit amount, in primary currency                 | NUMBER(18,4)    | 55          | N              |
| F_IVL_GRS_PRFT_AMT_LCL     | The total gross profit amount, in local currency.                  | NUMBER(18,4)    | 56          | N              |
| F_IVL_CUM_MKON_PCT         | The cumulative markon percent.                                     | NUMBER(12,4)    | 57          | N              |
| F_IVL_MKUP_AMT             | The value of upward revisions in price, in primary currency.       | NUMBER(18,4)    | 58          | N              |

| Name                      | Description                                                                                                     | Data Type/Bytes | Field order | Required field |
|---------------------------|-----------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_MKUP_AMT_LCL        | The value of upward revisions in price, in local currency.                                                      | NUMBER(18,4)    | 59          | N              |
| F_IVL_MKUP_CNCLLD_AMT     | The value of corrections to a upward revisions in price, in primary currency.                                   | NUMBER(18,4)    | 60          | N              |
| F_IVL_MKUP_CNCLLD_AMT_LCL | The value of corrections to a upward revisions in price, in local currency.                                     | NUMBER(18,4)    | 61          | N              |
| F_IVL_MKDN_CNCLLD_AMT     | The value of markdown cancellation to correct an unintentional error in a previous markup, in local currency.   | NUMBER(18,4)    | 62          | N              |
| F_IVL_MKDN_CNCLLD_AMT_LCL | The value of markdown cancellation to correct an unintentional error in a previous markup, in primary currency. | NUMBER(18,4)    | 63          | N              |
| F_IVL_PERM_MKDN_AMT       | The value of permanent reduction in price, in primary currency.                                                 | NUMBER(18,4)    | 64          | N              |
| F_IVL_PERM_MKDN_AMT_LCL   | The value of permanent reduction in price, in local currency.                                                   | NUMBER(18,4)    | 65          | N              |



| Name                     | Description                                                          | Data Type/Bytes | Field order | Required field |
|--------------------------|----------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_PRMTN_MKDN_AMT     | The value of promotion reductions of the price, in primary currency. | NUMBER(18,4)    | 66          | N              |
| F_IVL_PRMTN_MKDN_AMT_LCL | The value of promotion reductions of the price, in local currency.   | NUMBER(18,4)    | 67          | N              |
| F_IVL_CLRC_MKDN_AMT      | The value of clearance reductions of the price, in primary currency. | NUMBER(18,4)    | 68          | N              |
| F_IVL_CLRC_MKDN_AMT_LCL  | The value of clearance reductions of the price, in local currency.   | NUMBER(18,4)    | 69          | N              |
| F_IVL_EMPTY_DISC_AMT     | The value of employee discounts, in primary currency.                | NUMBER(18,4)    | 70          | N              |
| F_IVL_EMPTY_DISC_AMT_LCL | The value of employee discounts, in local currency.                  | NUMBER(18,4)    | 71          | N              |
| F_IVL_CASH_DISC_AMT      | The value of cash discounts, in primary currency.                    | NUMBER(18,4)    | 72          | N              |
| F_IVL_CASH_DISC_AMT_LCL  | The value of cash discounts, in local currency.                      | NUMBER(18,4)    | 73          | N              |
| F_IVL_FRGHT_COST_AMT     | The value of freight expenses, in primary currency.                  | NUMBER(18,4)    | 74          | N              |
| F_IVL_FRGHT_COST_AMT_LCL | The value of freight expenses, in local currency.                    | NUMBER(18,4)    | 75          | N              |

| Name                         | Description                                                                       | Data Type/Bytes | Field order | Required field |
|------------------------------|-----------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_WRKRM_COST_AMT         | The value of workroom expenses, in primary currency.                              | NUMBER(18,4)    | 76          | N              |
| F_IVL_WRKRM_COST_AMT_LCL     | The value of workroom expenses, in local currency                                 | NUMBER(18,4)    | 77          | N              |
| F_IVL_GAFS_COST_AMT          | The goods available for sale valued at cost, in primary currency.                 | NUMBER(18,4)    | 78          | N              |
| F_IVL_GAFS_COST_AMT_LCL      | The goods available for sale valued at cost, in local currency.                   | NUMBER(18,4)    | 79          | N              |
| F_IVL_GAFS_RTL_AMT           | The goods available for sale valued at retail, in primary currency.               | NUMBER(18,4)    | 80          | N              |
| F_IVL_GAFS_RTL_AMT_LCL       | The goods available for sale valued at retail, in local currency.                 | NUMBER(18,4)    | 81          | N              |
| F_IVL_SLS_QTY                | The number of net units of merchandise sold.                                      | NUMBER(12,4)    | 82          | N              |
| F_IVL_SLS_RTL_EX_VAT_AMT     | The value at retail, excluding VAT, of net merchandise sold, in primary currency. | NUMBER(18,4)    | 83          | N              |
| F_IVL_SLS_RTL_EX_VAT_AMT_LCL | The value at retail, excluding VAT, of net merchandise sold, in local currency.   | NUMBER(18,4)    | 84          | N              |

| Name                           | Description                                                                                      | Data Type/Bytes | Field order | Required field |
|--------------------------------|--------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_FRGHT_CLAIM_RTL_AMT      | The value at retail of freight claim, in primary currency.                                       | NUMBER(18,4)    | 85          | N              |
| F_IVL_FRGHT_CLAIM_RTL_AMT_LCL  | The value at retail of freight claim, in local currency.                                         | NUMBER(18,4)    | 86          | N              |
| F_IVL_FRGHT_CLAIM_COST_AMT     | The value at cost of freight claim, in primary currency.                                         | NUMBER(18,4)    | 87          | N              |
| F_IVL_FRGHT_CLAIM_COST_AMT_LCL | The value at cost of freight claim, in local currency.                                           | NUMBER(18,4)    | 88          | N              |
| F_IVL_IC_TSF_IN_COST_AMT       | The value at cost of inventory transferred in for intercompany transfers, in primary currency.   | NUMBER(18,4)    | 89          | N              |
| F_IVL_IC_TSF_IN_COST_AMT_LCL   | The value at cost of inventory transferred in for intercompany transfers, in local currency.     | NUMBER(18,4)    | 90          | N              |
| F_IVL_IC_TSF_IN_RT_L_AMT       | The value at retail of inventory transferred in for intercompany transfers, in primary currency. | NUMBER(18,4)    | 91          | N              |
| F_IVL_IC_TSF_IN_RT_L_AMT_LCL   | The value at retail of inventory transferred in for intercompany transfers, in local currency.   | NUMBER(18,4)    | 92          | N              |

| Name                              | Description                                                                                            | Data Type/Bytes | Field order | Required field |
|-----------------------------------|--------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_IC_TSF_OUT_C<br>OST_AMT     | The value at cost of inventory transferred out for intercompany transfers, in primary currency.        | NUMBER(18,4)    | 93          | N              |
| F_IVL_IC_TSF_OUT_C<br>OST_AMT_LCL | The value at cost of inventory transferred out for intercompany transfers, in local currency.          | NUMBER(18,4)    | 94          | N              |
| F_IVL_IC_TSF_OUT_R<br>TL_AMT      | The value at retail of inventory transferred out for intercompany transfers, in primary currency.      | NUMBER(18,4)    | 95          | N              |
| F_IVL_IC_TSF_OUT_R<br>TL_AMT_LCL  | The value at retail of inventory transferred out for intercompany transfers, in local currency.        | NUMBER(18,4)    | 96          | N              |
| F_IVL_IC_MARGIN_A<br>MT           | The margin value of intercompany transfers, in primary currency.                                       | NUMBER(18,4)    | 97          | N              |
| F_IVL_IC_MARGIN_A<br>MT_LCL       | The margin value of intercompany transfers, in local currency.                                         | NUMBER(18,4)    | 98          | N              |
| F_IVL_IC_MKDN_RTL<br>_AMT         | The markdown at retail of merchandise transferred out for intercompany transfers, in primary currency. | NUMBER(18,4)    | 99          | N              |

| Name                          | Description                                                                                                                        | Data Type/Bytes | Field order | Required field |
|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_IC_MKDN_RTL_AMT_LCL     | The markdown at retail of merchandise transferred out for intercompany transfers, in local currency.                               | NUMBER(18,4)    | 100         | N              |
| F_IVL_IC_MKUP_RTL_AMT         | The markup at retail of merchandise transferred out for intercompany transfers, in primary currency.                               | NUMBER(18,4)    | 101         | N              |
| F_IVL_IC_MKUP_RTL_AMT_LCL     | The markup at retail of merchandise transferred out for intercompany transfers, in local currency.                                 | NUMBER(18,4)    | 102         | N              |
| F_IVL_WO_UPD_INV_COST_AMT     | The value at cost of merchandise required work order activity, update inventory, for intercompany transfers, in primary currency.  | NUMBER(18,4)    | 103         | N              |
| F_IVL_WO_UPD_INV_COST_AMT_LCL | The value at cost of merchandise required work order activity, update inventory, for intercompany transfers, in local currency.    | NUMBER(18,4)    | 104         | N              |
| F_IVL_WO_POST_FIN_COST_AMT    | The value at cost of merchandise required work order activity, post to financial, for intercompany transfers, in primary currency. | NUMBER(18,4)    | 105         | N              |

| Name                           | Description                                                                                                                      | Data Type/Bytes | Field order | Required field |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_WO_POST_FIN_COST_AMT_LCL | The value at cost of merchandise required work order activity, post to financial, for intercompany transfers, in local currency. | NUMBER(18,4)    | 106         | N              |
| F_IVL_ADJ_COGS_COST_AMT        | The value at cost of stock adjustments that affect COGS, in primary currency.                                                    | NUMBER(18,4)    | 107         | N              |
| F_IVL_ADJ_COGS_COST_AMT_LCL    | The value at cost of stock adjustments that affect COGS, in local currency.                                                      | NUMBER(18,4)    | 108         | N              |
| F_IVL_ADJ_COGS_RT_L_AMT        | The value at retail of stock adjustments that affect COGS, in primary currency.                                                  | NUMBER(18,4)    | 109         | N              |
| F_IVL_ADJ_COGS_RT_L_AMT_LCL    | The value at retail of stock adjustments that affect COGS, in local currency.                                                    | NUMBER(18,4)    | 110         | N              |
| F_IVL_RESTOCK_FEE_AMT          | The value at cost of restocking fees received, in primary currency.                                                              | NUMBER(18,4)    | 111         | N              |
| F_IVL_RESTOCK_FEE_AMT_LCL      | The value at cost of restocking fees received, in local currency.                                                                | NUMBER(18,4)    | 112         | N              |
| F_IVL_DEAL_INCM_SLS_AMT        | The value of deal incomes sales received, in primary currency.                                                                   | NUMBER(18,4)    | 113         | N              |

| Name                          | Description                                                                                                                                                                                                                 | Data Type/Bytes | Field order | Required field |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_DEAL_INCM_SLS_AMT_LCL   | The value of deal incomes sales received, in local currency.                                                                                                                                                                | NUMBER(18,4)    | 114         | N              |
| F_IVL_DEAL_INCM_PURCH_AMT     | The value of deal incomes purchases received, in primary currency.                                                                                                                                                          | NUMBER(18,4)    | 115         | N              |
| F_IVL_DEAL_INCM_PURCH_AMT_LCL | The value of deal incomes purchases received, in local currency.                                                                                                                                                            | NUMBER(18,4)    | 116         | N              |
| F_IVL_COST_VAR_AMT            | The standard cost change as well as the cost difference between standard cost and transaction cost for transactions such as receiving, RTV and transfers using the standard cost method of accounting, in primary currency. | NUMBER(18,4)    | 117         | N              |
| F_IVL_COST_VAR_AMT_LCL        | The standard cost change as well as the cost difference between standard cost and transaction cost for transactions such as receiving, RTV and transfers using the standard cost method of accounting, in local currency.   | NUMBER(18,4)    | 118         | N              |

| Name                              | Description                                                                                 | Data Type/Bytes | Field order | Required field |
|-----------------------------------|---------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_RTL_COST_VA<br>R_AMT        | The cost variance using retail based accounting, in primary currency.                       | NUMBER(18,4)    | 119         | N              |
| F_IVL_RTL_COST_VA<br>R_AMT_LCL    | The cost variance using retail based accounting, in local currency.                         | NUMBER(18,4)    | 120         | N              |
| F_IVL_MARGIN_COS<br>T_VAR_AMT     | The cost variance using cost based accounting, in primary currency.                         | NUMBER(18,4)    | 121         | N              |
| F_IVL_MARGIN_COS<br>T_VAR_AMT_LCL | The cost variance using cost based accounting, in local currency.                           | NUMBER(18,4)    | 122         | N              |
| F_IVL_UP_CHRG_PRF<br>T_AMT        | The value of profit up charge costs incurred, in primary currency.                          | NUMBER(18,4)    | 123         | N              |
| F_IVL_UP_CHRG_PRF<br>T_AMT_LCL    | The value of expense up charge costs incurred, in primary currency.                         | NUMBER(18,4)    | 124         | N              |
| F_IVL_UP_CHRG_EXP<br>_AMT         | The value of expense up charge costs incurred, in primary currency.                         | NUMBER(18,4)    | 125         | N              |
| F_IVL_UP_CHRG_EXP<br>_AMT_LCL     | The value of expense up charge costs incurred, in local currency.                           | NUMBER(18,4)    | 126         | N              |
| F_IVL_TSF_IN_BK_C<br>OST_AMT      | The value at cost of inventory transferred in through a book transfer, in primary currency. | NUMBER(18,4)    | 127         | N              |



| Name                              | Description                                                                                    | Data Type/Bytes | Field order | Required field |
|-----------------------------------|------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_TSF_IN_BK_C<br>OST_AMT_LCL  | The value at cost of inventory transferred in through a book transfer, in local currency.      | NUMBER(18,4)    | 128         | N              |
| F_IVL_TSF_IN_BK_RT<br>L_AMT       | The value at retail of inventory transferred in through a book transfer, in primary currency.  | NUMBER(18,4)    | 129         | N              |
| F_IVL_TSF_IN_BK_RT<br>L_AMT_LCL   | The value at retail of inventory transferred in through a book transfer, in local currency.    | NUMBER(18,4)    | 130         | N              |
| F_IVL_TSF_OUT_BK_<br>COST_AMT     | The value at cost of inventory transferred out through a book transfer, in primary currency.   | NUMBER(18,4)    | 131         | N              |
| F_IVL_TSF_OUT_BK_<br>COST_AMT_LCL | The value at cost of inventory transferred out through a book transfer, in local currency.     | NUMBER(18,4)    | 132         | N              |
| F_IVL_TSF_OUT_BK_<br>RTL_AMT      | The value at retail of inventory transferred out through a book transfer, in primary currency. | NUMBER(18,4)    | 133         | N              |

| Name                         | Description                                                                                                                                                                                                           | Data Type/Bytes | Field order | Required field |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_TSF_OUT_BK_RTL_AMT_LCL | The value at retail of inventory transferred out through a book transfer, in local currency.                                                                                                                          | NUMBER(18,4)    | 134         | N              |
| F_IVL_INTER_STK_SL S_AMT     | The value of cumulative net sales since the last time a physical inventory was taken, in primary currency. It is valued at cost for the cost department and at retail for the retail department.                      | NUMBER(18,4)    | 135         | N              |
| F_IVL_INTER_STK_SL S_AMT_LCL | The cumulative net sales value since the last time a physical inventory was taken, in local currency. It is valued at cost for the cost department and at retail for the retail department.                           | NUMBER(18,4)    | 136         | N              |
| F_IVL_INTER_STK_S HRK_AMT    | The cumulative estimated (or budgeted) shrinkage value since the last time a physical inventory was taken, in primary currency. It is valued at cost for the cost department and at retail for the retail department. | NUMBER(18,4)    | 137         | N              |

| Name                             | Description                                                                                                                                                                                                         | Data Type/Bytes | Field order | Required field |
|----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_INTER_STK_S<br>HRK_AMT_LCL | The cumulative estimated (or budgeted) shrinkage value since the last time a physical inventory was taken, in local currency. It is valued at cost for the cost department and at retail for the retail department. | NUMBER(18,4)    | 138         | N              |
| F_IVL_STK_MTD_SLS<br>_AMT        | The month-to-date net sales value, in primary currency. It is valued at cost for the cost department and at retail for the retail department.                                                                       | NUMBER(18,4)    | 139         | N              |
| F_IVL_STK_MTD_SLS<br>_AMT_LCL    | The month-to-date net sales value, in local currency. It is valued at cost for the cost department and at retail for the retail department.                                                                         | NUMBER(18,4)    | 140         | N              |
| F_IVL_STK_MTD_SH<br>RK_AMT       | The month-to-date estimated (or budgeted) shrinkage value, in primary currency. It is valued at cost for the cost department and at retail for the retail department.                                               | NUMBER(18,4)    | 141         | N              |

| Name                          | Description                                                                                                                                                         | Data Type/Bytes | Field order | Required field |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_STK_MTD_SHRK_AMT_LCL    | The month-to-date estimated (or budgeted) shrinkage value, in local currency. It is valued at cost for the cost department and at retail for the retail department. | NUMBER(18,4)    | 142         | N              |
| F_IVL_BK_STOCK_RT_L_AMT       | The value at retail of book stock, in primary currency.                                                                                                             | NUMBER(18,4)    | 143         | N              |
| F_IVL_BK_STOCK_RT_L_AMT_LCL   | The value at retail of book stock, in local currency.                                                                                                               | NUMBER(18,4)    | 144         | N              |
| F_IVL_BK_STOCK_COST_AMT       | The value at cost of book stock, in primary currency.                                                                                                               | NUMBER(18,4)    | 145         | N              |
| F_IVL_BK_STOCK_COST_AMT_LCL   | The value at cost of book stock, in local currency.                                                                                                                 | NUMBER(18,4)    | 146         | N              |
| F_IVL_ACTL_STOCK_COST_AMT     | The value at cost of actual stock, when the physical inventory is taken, in primary currency.                                                                       | NUMBER(18,4)    | 147         | N              |
| F_IVL_ACTL_STOCK_COST_AMT_LCL | The value at cost of actual stock, when the physical inventory is taken, in local currency.                                                                         | NUMBER(18,4)    | 148         | N              |

| Name                         | Description                                                                                     | Data Type/Bytes | Field order | Required field |
|------------------------------|-------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_ACTL_STOCK_RTL_AMT     | The value at retail of actual stock, when the physical inventory is taken, in primary currency. | NUMBER(18,4)    | 149         | N              |
| F_IVL_ACTL_STOCK_RTL_AMT_LCL | The value at retail of actual stock, when the physical inventory is taken, in local currency.   | NUMBER(18,4)    | 150         | N              |

**stlblwdm.txt**

Business rules:

- This interface file contains stock ledger values for a department, class, subclass and location on a given week.
- This interface file cannot contain duplicate transactions for a dept\_idnt, class\_idnt, subclass\_idnt, loc\_idnt and day\_dt combination.
- This interface file follows the fact flat file interface layout standard.
- For this interface file, the day\_dt represents the end day of a week.
- This interface file does not need to be provided when the stock ledger uses Gregorian time (because this table is not populated).

| Name         | Description                                                     | Data Type/Bytes | Field order | Required field |
|--------------|-----------------------------------------------------------------|-----------------|-------------|----------------|
| SBCLASS_IDNT | The unique identifier of the subclass in the product hierarchy. | CHARACTER(4)    | 1           | Yes            |
| CLASS_IDNT   | The unique identifier of the class in the product hierarchy.    | CHARACTER(4)    | 2           | Yes            |

| Name                       | Description                                                              | Data Type/Bytes | Field order | Required field |
|----------------------------|--------------------------------------------------------------------------|-----------------|-------------|----------------|
| DEPT_IDNT                  | The unique identifier of a department in the product hierarchy.          | CHARACTER(4)    | 3           | Yes            |
| LOC_IDNT                   | The unique identifier of the location.                                   | CHARACTER(10)   | 4           | Yes            |
| LOC_TYPE_CDE               | The code that indicates whether the location is a store or warehouse.    | CHARACTER(2)    | 5           | Yes            |
| DAY_DT                     | The calendar day on which the transaction occurred.                      | DATE            | 6           | Yes            |
| F_IVL_BEG_SOH_COST_AMT     | The beginning of period stock on hand total cost, in primary currency.   | NUMBER(18,4)    | 7           | No             |
| F_IVL_BEG_SOH_COST_AMT_LCL | The beginning of period stock on hand total cost, in local currency.     | NUMBER(18,4)    | 8           | No             |
| F_IVL_BEG_SOH_RTL_AMT      | The beginning of period stock on hand total retail, in primary currency. | NUMBER(18,4)    | 9           | No             |

| Name                       | Description                                                            | Data Type/Bytes | Field order | Required field |
|----------------------------|------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_BEG_SOH_RTL_AMT_LCL  | The beginning of period stock on hand total retail, in local currency. | NUMBER(18,4)    | 10          | No             |
| F_IVL_SOH_ADJ_COST_AMT     | The value at cost of stock on hand adjustments, in primary currency.   | NUMBER(18,4)    | 11          | No             |
| F_IVL_SOH_ADJ_COST_AMT_LCL | The value at cost of stock on hand adjustments, in local currency.     | NUMBER(18,4)    | 12          | No             |
| F_IVL_SOH_ADJ_RTL_AMT      | The value at retail of stock on hand adjustments, in primary currency. | NUMBER(18,4)    | 13          | No             |
| F_IVL_SOH_ADJ_RTL_AMT_LCL  | The value at retail of stock on hand adjustments, in local currency.   | NUMBER(18,4)    | 14          | No             |
| F_IVL_RCPTS_COST_AMT       | The value at cost of inventory received, in primary currency.          | NUMBER(18,4)    | 15          | No             |
| F_IVL_RCPTS_COST_AMT_LCL   | The value at cost of inventory received, in local currency.            | NUMBER(18,4)    | 16          | No             |

| Name                    | Description                                                                 | Data Type/Bytes | Field order | Required field |
|-------------------------|-----------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_RCPTS_RTL_AMT     | The value at retail of inventory received, in primary currency.             | NUMBER(18,4)    | 17          | No             |
| F_IVL_RCPTS_RTL_AMT_LCL | The value at retail of inventory received, in local currency.               | NUMBER(18,4)    | 18          | No             |
| F_IVL_RTV_COST_AMT      | The value at cost of inventory returned to a vendor, in primary currency.   | NUMBER(18,4)    | 19          | No             |
| F_IVL_RTV_COST_AMT_LCL  | The value at cost of inventory returned to a vendor, in local currency.     | NUMBER(18,4)    | 20          | No             |
| F_IVL_RTV_RTL_AMT       | The value at retail of inventory returned to a vendor, in primary currency. | NUMBER(18,4)    | 21          | No             |
| F_IVL_RTV_RTL_AMT_LCL   | The value at retail of inventory returned to a vendor, in local currency.   | NUMBER(18,4)    | 22          | No             |



| Name                       | Description                                                            | Data Type/Bytes | Field order | Required field |
|----------------------------|------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_TSF_IN_COST_AMT      | The value at cost of inventory transferred in, in primary currency.    | NUMBER(18,4)    | 23          | No             |
| F_IVL_TSF_IN_COST_AMT_LCL  | The value at cost of inventory transferred in, in local currency.      | NUMBER(18,4)    | 24          | No             |
| F_IVL_TSF_IN_RTL_AMT       | The value at retail of inventory transferred in, in primary currency.  | NUMBER(18,4)    | 25          | No             |
| F_IVL_TSF_IN_RTL_AMT_LCL   | The value at retail of inventory transferred in, in local currency.    | NUMBER(18,4)    | 26          | No             |
| F_IVL_TSF_OUT_COST_AMT     | The value at cost of inventory transferred out, in primary currency.   | NUMBER(18,4)    | 27          | No             |
| F_IVL_TSF_OUT_COST_AMT_LCL | The value at cost of inventory transferred out, in local currency.     | NUMBER(18,4)    | 28          | No             |
| F_IVL_TSF_OUT_RTL_AMT      | The value at retail of inventory transferred out, in primary currency. | NUMBER(18,4)    | 29          | No             |

| Name                      | Description                                                                                     | Data Type/Bytes | Field order | Required field |
|---------------------------|-------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_TSF_OUT_RTL_AMT_LCL | The value at retail of inventory transferred out, in local currency.                            | NUMBER(18,4)    | 30          | No             |
| F_IVL_SHRK_COST_AMT       | The value at cost of the difference between actual and ending inventory, in primary currency.   | NUMBER(18,4)    | 31          | No             |
| F_IVL_SHRK_COST_AMT_LCL   | The value at cost of the difference between actual and ending inventory, in local currency.     | NUMBER(18,4)    | 32          | No             |
| F_IVL_SHRK_RTL_AMT        | The value at retail of the difference between actual and ending inventory, in primary currency. | NUMBER(18,4)    | 33          | No             |
| F_IVL_SHRK_RTL_AMT_LCL    | The value at retail of the difference between actual and ending inventory, in local currency.   | NUMBER(18,4)    | 34          | No             |

| Name                          | Description                                                                        | Data Type/Bytes | Field order | Required field |
|-------------------------------|------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_RTRNS_COST_AMT          | The value at cost of inventory returned from sales, in primary currency.           | NUMBER(18,4)    | 35          | No             |
| F_IVL_RTRNS_COST_AMT_LCL      | The value at cost of inventory returned from sales, in local currency.             | NUMBER(18,4)    | 36          | No             |
| F_IVL_RTRNS_RTL_AMT           | The value at retail of inventory returned from sales, in primary currency.         | NUMBER(18,4)    | 37          | No             |
| F_IVL_RTRNS_RTL_AMT_LCL       | The value at retail of inventory returned from sales, in local currency.           | NUMBER(18,4)    | 38          | No             |
| F_IVL_RECLASS_IN_COST_AMT     | The value at cost of inventory reclassified to this location, in primary currency. | NUMBER(18,4)    | 39          | No             |
| F_IVL_RECLASS_IN_COST_AMT_LCL | The value at cost of inventory reclassified to this location, in local currency.   | NUMBER(18,4)    | 40          | No             |

| Name                           | Description                                                                            | Data Type/Bytes | Field order | Required field |
|--------------------------------|----------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_RECLASS_IN_RTL_AMT       | The value at retail of inventory reclassified to this location, in primary currency.   | NUMBER(18,4)    | 41          | No             |
| F_IVL_RECLASS_IN_RTL_AMT_LCL   | The value at retail of inventory reclassified to this location, in local currency.     | NUMBER(18,4)    | 42          | No             |
| F_IVL_RECLASS_OUT_COST_AMT     | The value at cost of inventory reclassified from this location, in primary currency.   | NUMBER(18,4)    | 43          | No             |
| F_IVL_RECLASS_OUT_COST_AMT_LCL | The value at cost of inventory reclassified from this location, in local currency.     | NUMBER(18,4)    | 44          | No             |
| F_IVL_RECLASS_OUT_RTL_AMT      | The value at retail of inventory reclassified from this location, in primary currency. | NUMBER(18,4)    | 45          | No             |

| Name                          | Description                                                                          | Data Type/Bytes | Field order | Required field |
|-------------------------------|--------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_RECLASS_OUT_RTL_AMT_LCL | The value at retail of inventory reclassified from this location, in local currency. | NUMBER(18,4)    | 46          | No             |
| F_IVL_SLS_COST_AMT            | The value at cost of inventory sold, in primary currency.                            | NUMBER(18,4)    | 47          | No             |
| F_IVL_SLS_COST_AMT_LCL        | The value at cost of inventory sold, in local currency.                              | NUMBER(18,4)    | 48          | No             |
| F_IVL_SLS_RTL_AMT             | The value at retail of inventory sold, in primary currency.                          | NUMBER(18,4)    | 49          | No             |
| F_IVL_SLS_RTL_AMT_LCL         | The value at retail of inventory sold, in local currency.                            | NUMBER(18,4)    | 50          | No             |
| F_IVL_END_SOH_COST_AMT        | The end of period stock on hand total cost, in primary currency.                     | NUMBER(18,4)    | 51          | No             |
| F_IVL_END_SOH_COST_AMT_LCL    | The end of period stock on hand total cost, in local currency.                       | NUMBER(18,4)    | 52          | No             |

| <b>Name</b>                  | <b>Description</b>                                                                             | <b>Data Type/Bytes</b> | <b>Field order</b> | <b>Required field</b> |
|------------------------------|------------------------------------------------------------------------------------------------|------------------------|--------------------|-----------------------|
| F_IVL_END_SOH_RTL_AMT        | The end of period stock on hand total retail, in primary currency.                             | NUMBER(18,4)           | 53                 | No                    |
| F_IVL_END_SOH_RTL_AMT_LCL    | The end of period stock on hand total retail, in local currency.                               | NUMBER(18,4)           | 54                 | No                    |
| F_IVL_GRS_PRFT_AMT           | The total gross profit amount, in primary currency.                                            | NUMBER(18,4)           | 55                 | No                    |
| F_IVL_GRS_PRFT_AMT_LCL       | The total gross profit amount, in local currency.                                              | NUMBER(18,4)           | 56                 | No                    |
| F_IVL_CUM_MKON_PCT           | The cumulative markon percent.                                                                 | NUMBER(12,4)           | 57                 | No                    |
| F_IVL_ADJ_STOCK_COST_AMT     | The value at cost of adjusted stock when the physical inventory is taken, in primary currency. | NUMBER(18,4)           | 58                 | No                    |
| F_IVL_ADJ_STOCK_COST_AMT_LCL | The value at cost of adjusted stock when the physical inventory is taken, in local currency.   | NUMBER(18,4)           | 59                 | No                    |

| Name                        | Description                                                                                      | Data Type/Bytes | Field order | Required field |
|-----------------------------|--------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_ADJ_STOCK_RTL_AMT     | The value at retail of adjusted stock when the physical inventory is taken, in primary currency. | NUMBER(18,4)    | 60          | No             |
| F_IVL_ADJ_STOCK_RTL_AMT_LCL | The value at retail of adjusted stock when the physical inventory is taken, in local currency.   | NUMBER(18,4)    | 61          | No             |
| F_IVL_MKUP_AMT              | The value of upward revisions in price, in primary currency.                                     | NUMBER(18,4)    | 62          | No             |
| F_IVL_MKUP_AMT_LCL          | The value of upward revisions in price, in local currency.                                       | NUMBER(18,4)    | 63          | No             |
| F_IVL_MKUP_CNCLLD_AMT       | The value of corrections to upward revisions in price, in primary currency.                      | NUMBER(18,4)    | 64          | No             |
| F_IVL_MKUP_CNCLLD_AMT_LCL   | The value of corrections to upward revisions in price, in local currency.                        | NUMBER(18,4)    | 65          | No             |

| Name                      | Description                                                                                                     | Data Type/Bytes | Field order | Required field |
|---------------------------|-----------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_MKDN_CNCLLD_AMT     | The value of markdown cancellation to correct an unintentional error in a previous markup, in primary currency. | NUMBER(18,4)    | 66          | No             |
| F_IVL_MKDN_CNCLLD_AMT_LCL | The value of markdown cancellation to correct an unintentional error in a previous markup, in local currency.   | NUMBER(18,4)    | 67          | No             |
| F_IVL_PERM_MKDN_AMT       | The value of permanent reductions of the price, in primary currency.                                            | NUMBER(18,4)    | 68          | No             |
| F_IVL_PERM_MKDN_AMT_LCL   | The value of permanent reductions of the price, in local currency.                                              | NUMBER(18,4)    | 69          | No             |
| F_IVL_PRMTN_MKDN_AMT      | The value of promotion reductions of the price, in primary currency.                                            | NUMBER(18,4)    | 70          | No             |
| F_IVL_PRMTN_MKDN_AMT_LCL  | The value of promotion reductions of the price, in local currency.                                              | NUMBER(18,4)    | 71          | No             |



| Name                     | Description                                                          | Data Type/Bytes | Field order | Required field |
|--------------------------|----------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_CLRC_MKDN_AMT      | The value of clearance reductions of the price, in primary currency. | NUMBER(18,4)    | 72          | No             |
| F_IVL_CLRC_MKDN_AMT_LCL  | The value of clearance reductions of the price, in local currency    | NUMBER(18,4)    | 73          | No             |
| F_IVL_EMPTY_DISC_AMT     | The value of employee discounts, in primary currency.                | NUMBER(18,4)    | 74          | No             |
| F_IVL_EMPTY_DISC_AMT_LCL | The value of employee discounts, in local currency.                  | NUMBER(18,4)    | 75          | No             |
| F_IVL_CASH_DISC_AMT      | The value of cash discounts, in primary currency.                    | NUMBER(18,4)    | 76          | No             |
| F_IVL_CASH_DISC_AMT_LCL  | The value of cash discounts, in local currency.                      | NUMBER(18,4)    | 77          | No             |
| F_IVL_FRGHT_COST_AMT     | The value of freight expenses, in primary currency.                  | NUMBER(18,4)    | 78          | No             |
| F_IVL_FRGHT_COST_AMT_LCL | The value of freight expenses, in local currency.                    | NUMBER(18,4)    | 79          | No             |

| Name                     | Description                                                         | Data Type/Bytes | Field order | Required field |
|--------------------------|---------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_WRKRM_COST_AMT     | The value of workroom expenses, in primary currency.                | NUMBER(18,4)    | 80          | No             |
| F_IVL_WRKRM_COST_AMT_LCL | The value of workroom expenses, in local currency.                  | NUMBER(18,4)    | 81          | No             |
| F_IVL_GAFS_COST_AMT      | The goods available for sale valued at cost, in primary currency.   | NUMBER(18,4)    | 82          | No             |
| F_IVL_GAFS_COST_AMT_LCL  | The goods available for sale valued at cost, in local currency.     | NUMBER(18,4)    | 83          | No             |
| F_IVL_GAFS_RTL_AMT       | The goods available for sale valued at retail, in primary currency. | NUMBER(18,4)    | 84          | No             |
| F_IVL_GAFS_RTL_AMT_LCL   | The goods available for sale valued at retail, in local currency.   | NUMBER(18,4)    | 85          | No             |
| F_IVL_SLS_QTY            | The number of net units of merchandise sold.                        | NUMBER(12,4)    | 86          | No             |

| Name                           | Description                                                                                    | Data Type/Bytes | Field order | Required field |
|--------------------------------|------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_SLS_RTL_EX_VAT_AMT       | The value at retail, excluding VAT, of net merchandise sold, in primary currency.              | NUMBER(18,4)    | 87          | No             |
| F_IVL_SLS_RTL_EX_VAT_AMT_LCL   | The value at retail, excluding VAT, of net merchandise sold, in local currency.                | NUMBER(18,4)    | 88          | No             |
| F_IVL_FRGHT_CLAIM_RTL_AMT      | The value at retail of freight claim, in primary currency.                                     | NUMBER(18,4)    | 89          | No             |
| F_IVL_FRGHT_CLAIM_RTL_AMT_LCL  | The value at retail of freight claim, in local currency.                                       | NUMBER(18,4)    | 90          | No             |
| F_IVL_FRGHT_CLAIM_COST_AMT     | The value at cost of freight claim, in primary currency.                                       | NUMBER(18,4)    | 91          | No             |
| F_IVL_FRGHT_CLAIM_COST_AMT_LCL | The value at cost of freight claim, in local currency.                                         | NUMBER(18,4)    | 92          | No             |
| F_IVL_IC_TSF_IN_COST_AMT       | The value at cost of inventory transferred in for intercompany transfers, in primary currency. | NUMBER(18,4)    | 93          | No             |

| Name                          | Description                                                                                      | Data Type/Bytes | Field order | Required field |
|-------------------------------|--------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_IC_TSF_IN_COST_AMT_LCL  | The value at cost of inventory transferred in for intercompany transfers, in local currency.     | NUMBER(18,4)    | 94          | No             |
| F_IVL_IC_TSF_IN_RTL_AMT       | The value at retail of inventory transferred in for intercompany transfers, in primary currency. | NUMBER(18,4)    | 95          | No             |
| F_IVL_IC_TSF_IN_RTL_AMT_LCL   | The value at retail of inventory transferred in for intercompany transfers, in local currency.   | NUMBER(18,4)    | 96          | No             |
| F_IVL_IC_TSF_OUT_COST_AMT     | The value at cost of inventory transferred out for intercompany transfers, in primary currency.  | NUMBER(18,4)    | 97          | No             |
| F_IVL_IC_TSF_OUT_COST_AMT_LCL | The value at cost of inventory transferred out for intercompany transfers, in local currency.    | NUMBER(18,4)    | 98          | No             |

| Name                         | Description                                                                                            | Data Type/Bytes | Field order | Required field |
|------------------------------|--------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_IC_TSF_OUT_RTL_AMT     | The value at retail of inventory transferred out for intercompany transfers, in primary currency.      | NUMBER(18,4)    | 99          | No             |
| F_IVL_IC_TSF_OUT_RTL_AMT_LCL | The value at retail of inventory transferred out for intercompany transfers, in local currency.        | NUMBER(18,4)    | 100         | No             |
| F_IVL_IC_MARGIN_AMT          | The margin value of intercompany transfers, in primary currency.                                       | NUMBER(18,4)    | 101         | No             |
| F_IVL_IC_MARGIN_AMT_LCL      | The margin value of intercompany transfers, in local currency.                                         | NUMBER(18,4)    | 102         | No             |
| F_IVL_IC_MKDN_RTL_AMT        | The markdown at retail of merchandise transferred out for intercompany transfers, in primary currency. | NUMBER(18,4)    | 103         | No             |

| Name                      | Description                                                                                                                       | Data Type/Bytes | Field order | Required field |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_IC_MKDN_RTL_AMT_LCL | The markdown at retail of merchandise transferred out for intercompany transfers, in local currency.                              | NUMBER(18,4)    | 104         | No             |
| F_IVL_IC_MKUP_RTL_AMT     | The markup at retail of merchandise transferred out for intercompany transfers, in primary currency.                              | NUMBER(18,4)    | 105         | No             |
| F_IVL_IC_MKUP_RTL_AMT_LCL | The markup at retail of merchandise transferred out for intercompany transfers, in local currency.                                | NUMBER(18,4)    | 106         | No             |
| F_IVL_WO_UPD_INV_COST_AMT | The value at cost of merchandise required work order activity, update inventory, for intercompany transfers, in primary currency. | NUMBER(18,4)    | 107         | No             |

| Name                           | Description                                                                                                                        | Data Type/Bytes | Field order | Required field |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_WO_UPD_INV_COST_AMT_LCL  | The value at cost of merchandise required work order activity, update inventory, for intercompany transfers, in local currency.    | NUMBER(18,4)    | 108         | No             |
| F_IVL_WO_POST_FIN_COST_AMT     | The value at cost of merchandise required work order activity, post to financial, for intercompany transfers, in primary currency. | NUMBER(18,4)    | 109         | No             |
| F_IVL_WO_POST_FIN_COST_AMT_LCL | The value at cost of merchandise required work order activity, post to financial, for intercompany transfers, in local currency.   | NUMBER(18,4)    | 110         | No             |
| F_IVL_ADJ_COGS_COST_AMT        | The value at cost of stock adjustments that affect COGS, in primary currency.                                                      | NUMBER(18,4)    | 111         | No             |

| Name                        | Description                                                                    | Data Type/Bytes | Field order | Required field |
|-----------------------------|--------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_ADJ_COGS_COST_AMT_LCL | The value at cost of stock adjustments that affect COGS, in local currency.    | NUMBER(18,4)    | 112         | No             |
| F_IVL_ADJ_COGS_RTL_AMT      | The value at retail of stock adjustments that affect COGS, in primary currency | NUMBER(18,4)    | 113         | No             |
| F_IVL_ADJ_COGS_RTL_AMT_LCL  | The value at retail of stock adjustments that affect COGS, in local currency   | NUMBER(18,4)    | 114         | No             |
| F_IVL_RESTOCK_FEE_AMT       | The value at cost of restocking fees received, in primary currency.            | NUMBER(18,4)    | 115         | No             |
| F_IVL_RESTOCK_FEE_AMT_LCL   | The value at cost of restocking fees received, in local currency.              | NUMBER(18,4)    | 116         | No             |
| F_IVL_DEAL_INCM_SLS_AMT     | The value of deal incomes sales received, in primary currency.                 | NUMBER(18,4)    | 117         | No             |



| Name                          | Description                                                                                                                                                                                                                 | Data Type/Bytes | Field order | Required field |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_DEAL_INCM_SLS_AMT_LCL   | The value of deal incomes sales received, in local currency.                                                                                                                                                                | NUMBER(18,4)    | 118         | No             |
| F_IVL_DEAL_INCM_PURCH_AMT     | The value of deal incomes purchases received, in primary currency.                                                                                                                                                          | NUMBER(18,4)    | 119         | No             |
| F_IVL_DEAL_INCM_PURCH_AMT_LCL | The value of deal incomes purchases received, in local currency.                                                                                                                                                            | NUMBER(18,4)    | 120         | No             |
| F_IVL_COST_VAR_AMT            | The standard cost change as well as the cost difference between standard cost and transaction cost for transactions such as receiving, RTV and transfers using the standard cost method of accounting, in primary currency. | NUMBER(18,4)    | 121         | No             |

| Name                       | Description                                                                                                                                                                                                               | Data Type/Bytes | Field order | Required field |
|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_COST_VAR_AMT_LCL     | The standard cost change as well as the cost difference between standard cost and transaction cost for transactions such as receiving, RTV and transfers using the standard cost method of accounting, in local currency. | NUMBER(18,4)    | 122         | No             |
| F_IVL_RTL_COST_VAR_AMT     | The cost variance using retail based accounting, in primary currency.                                                                                                                                                     | NUMBER(18,4)    | 123         | No             |
| F_IVL_RTL_COST_VAR_AMT_LCL | The cost variance using retail based accounting, in local currency.                                                                                                                                                       | NUMBER(18,4)    | 124         | No             |
| F_IVL_MARGIN_COST_VAR_AMT  | The cost variance using cost based accounting, in primary currency.                                                                                                                                                       | NUMBER(18,4)    | 125         | No             |

| Name                          | Description                                                                                 | Data Type/Bytes | Field order | Required field |
|-------------------------------|---------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_MARGIN_COST_VAR_AMT_LCL | The cost variance using cost based accounting, in local currency.                           | NUMBER(18,4)    | 126         | No             |
| F_IVL_UP_CHRG_PRFT_AMT        | The value of profit up charge costs incurred, in primary currency.                          | NUMBER(18,4)    | 127         | No             |
| F_IVL_UP_CHRG_PRFT_AMT_LCL    | The value of profit up charge costs incurred, in local currency.                            | NUMBER(18,4)    | 128         | No             |
| F_IVL_UP_CHRG_EXP_AMT         | The value of expense up charge costs incurred, in primary currency.                         | NUMBER(18,4)    | 129         | No             |
| F_IVL_UP_CHRG_EXP_AMT_LCL     | The value of expense up charge costs incurred, in local currency.                           | NUMBER(18,4)    | 130         | No             |
| F_IVL_TSF_IN_BK_COST_AMT      | The value at cost of inventory transferred in through a book transfer, in primary currency. | NUMBER(18,4)    | 131         | No             |

| Name                          | Description                                                                                   | Data Type/Bytes | Field order | Required field |
|-------------------------------|-----------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_TSF_IN_BK_COST_AMT_LCL  | The value at cost of inventory transferred in through a book transfer, in local currency.     | NUMBER(18,4)    | 132         | No             |
| F_IVL_TSF_IN_BK_RTL_AMT       | The value at retail of inventory transferred in through a book transfer, in primary currency. | NUMBER(18,4)    | 133         | No             |
| F_IVL_TSF_IN_BK_RTL_AMT_LCL   | The value at retail of inventory transferred in through a book transfer, in local currency.   | NUMBER(18,4)    | 134         | No             |
| F_IVL_TSF_OUT_BK_COST_AMT     | The value at cost of inventory transferred out through a book transfer, in primary currency.  | NUMBER(18,4)    | 135         | No             |
| F_IVL_TSF_OUT_BK_COST_AMT_LCL | The value at cost of inventory transferred out through a book transfer, in local currency.    | NUMBER(18,4)    | 136         | No             |

| Name                             | Description                                                                                    | Data Type/Bytes | Field order | Required field |
|----------------------------------|------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_IVL_TSF_OUT_BK_RTL_AMT         | The value at retail of inventory transferred out through a book transfer, in primary currency. | NUMBER(18,4)    | 137         | No             |
| F_IVL_TSF_OUT_BK_RTL_AMT_LC<br>L | The value at retail of inventory transferred out through a book transfer, in local currency.   | NUMBER(18,4)    | 138         | No             |

### subtrantypedm.txt

Business rules:

- This interface file contains sub-transaction type records.
- This interface file cannot contain duplicate records for a sub\_tran\_type\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name               | Description                                        | Data Type/Bytes | Field order | Required field |
|--------------------|----------------------------------------------------|-----------------|-------------|----------------|
| SUB_TRAN_TYPE_IDNT | The unique identifier of the sub-transaction type. | VARCHAR2(6)     | 1           | Yes            |
| SUB_TRAN_TYPE_DESC | The description of the sub-transaction type.       | VARCHAR2(120)   | 2           | No             |

### supctrdm.txt

Business rules:

- This interface file contains supplier contract information for status in 'A', 'C', 'X'.
- This interface file cannot contain duplicate records for a cntct\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name                  | Description                                                                                       | Data Type/Bytes | Field order | Required field |
|-----------------------|---------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| CNTRCT_IDNT           | The unique identifier of a contract.                                                              | CHARACTER(6)    | 1           | Yes            |
| SUPP_IDNT             | The unique identifier of a supplier.                                                              | CHARACTER(10)   | 2           | Yes            |
| STATUS_CDE            | The code representing the status for this contract.                                               | VARCHAR2(1)     | 3           | Yes            |
| CNTRCT_BEG_DT         | The starting date for the contract.                                                               | DATE            | 4           | No             |
| CNTRCT_END_DT         | The ending date for the contract.                                                                 | DATE            | 5           | No             |
| CNTRCT_DIST           | The distributor name who collects the merchandise from the supplier and delivers to the retailer. | VARCHAR2(40)    | 6           | No             |
| CNTRCT_SHIP_MTHD_CDE  | The code representing the method of shipment associated with the contract.                        | VARCHAR2(2)     | 7           | No             |
| CNTRCT_SHIP_MTHD_DESC | The description of the method of shipment associated with the contract.                           | VARCHAR2(120)   | 8           | No             |
| STATUS_DESC           | The description of the contract status.                                                           | VARCHAR2(120)   | 9           | No             |

**supsupdm.txt**

Business rules:

- This interface file contains a record for each supplier, and it holds details of supplier related attributes.
- This interface file cannot contain duplicate records for a supp\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name                | Description                                                                  | Data Type/Bytes | Field order | Required field |
|---------------------|------------------------------------------------------------------------------|-----------------|-------------|----------------|
| SUPP_IDNT           | The unique identifier of a supplier.                                         | VARCHAR2(10)    | 1           | Yes            |
| SUPP_DESC           | The supplier's name.                                                         | VARCHAR2(240)   | 2           | Yes            |
| SUPP_SECND_DESC     | The supplier's second name                                                   | VARCHAR2(240)   | 3           | No             |
| SUPP_QC_RQRD_IND    | Indicates if this supplier's receipts should be checked for quality control. | VARCHAR2(1)     | 4           | No             |
| SUPP_PRE_MARK_IND   | Indicates whether the items supplied by this supplier is pre-marked.         | VARCHAR2(1)     | 5           | No             |
| SUPP_PRE_TICKET_IND | Indicates if the supplier pre-marks or pre-prices his goods.                 | VARCHAR2(1)     | 6           | No             |
| SUPP_STTS_CDE       | The code that indicates if the supplier is currently active.                 | VARCHAR2(2)     | 7           | No             |
| SUPP_STTS_DESC      | The description of the status code.                                          | VARCHAR2(120)   | 8           | No             |
| SUPP_EDI_IND        | This column indicates if the supplier has EDI capabilities.                  | VARCHAR2(1)     | 9           | No             |
| SUPP_DOMESTIC_CDE   | Supplier's domestic code.                                                    | VARCHAR2(1)     | 10          | No             |
| SUPP_DOMESTIC_DESC  | The description of the supplier's domestic code.                             | VARCHAR2(120)   | 11          | No             |

| Name            | Description                                                          | Data Type/Bytes | Field order | Required field |
|-----------------|----------------------------------------------------------------------|-----------------|-------------|----------------|
| SUPP_CRNCY_CDE  | The code representing the currency that the supplier operates under. | VARCHAR2(3)     | 12          | No             |
| SUPP_CRNCY_DESC | The description of the supplier's currency code.                     | VARCHAR2(120)   | 13          | No             |
| SUPP_VMI_IND    | Indicates whether a supplier is vendor managed inventory supplier.   | VARCHAR2(1)     | 14          | No             |

**suptrmdm.txt**

Business rules:

- This interface file defines the associations between supplier and supplier trait.
- This interface file cannot contain duplicate records for a supp\_trait\_idnt, supp\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name            | Description                                  | Data Type/Bytes | Field order | Required field |
|-----------------|----------------------------------------------|-----------------|-------------|----------------|
| SUPP_TRAIT_IDNT | The unique identifier of the supplier trait. | VARCHAR2(10)    | 1           | Yes            |
| SUPP_IDNT       | The unique identifier of a supplier.         | VARCHAR2(10)    | 2           | Yes            |

**suptrtdm.txt**

Business rules:

- This interface file contains supplier trait information.
- This interface file cannot contain duplicate records for a supp\_trait\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name            | Description                                  | Data Type/Bytes | Field order | Required field |
|-----------------|----------------------------------------------|-----------------|-------------|----------------|
| SUPP_TRAIT_IDNT | The unique identifier of the supplier trait. | VARCHAR2(10)    | 1           | Yes            |



| Name            | Description                                                                                 | Data Type/Bytes | Field order | Required field |
|-----------------|---------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| MAST_SUPP_FLAG  | Flag which indicates if this trait is a master supplier trait. Valid values are 'Y' or 'N'. | VARCHAR2(1)     | 2           | Yes            |
| SUPP_TRAIT_DESC | The supplier trait description.                                                             | VARCHAR2(120)   | 3           | No             |
| MAST_SUPP_CDE   | The number of the master supplier.                                                          | VARCHAR2(10)    | 4           | No             |

**tndrtypdm.txt**

Business rules:

- This interface file contains tender types and their parent tender type groups.
- This interface file cannot contain duplicate records for a tndr\_type\_id\_idnt, tndr\_type\_grp\_idnt combination.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name               | Description                                                                                                                                    | Data Type/Bytes | Field order | Required field |
|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| TNDR_TYPE_GRP_IDNT | The unique identifier for the tender type group. . An example of a tender type group is cash, check, or credit card.                           | VARCHAR2(6)     | 1           | Yes            |
| TNDR_TYPE_ID_IDNT  | The unique identifier for the tender type ID within a tender type group. An example of a tender type ID is Discover Card, Master Card, or Visa | VARCHAR2(6)     | 2           | Yes            |
| TNDR_TYPE_GRP_DESC | The description of the tender type group. An example of the description may be "Credit Cards", "Cash", or "Check".                             | VARCHAR2(120)   | 3           | No             |

| Name              | Description                                                                                                                                | Data Type/Bytes | Field order | Required field |
|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| TNDR_TYPE_ID_DESC | The description of the tender type ID. An example of the ID description may be "Master Card", "Visa Gold", or American Express Corporate". | VARCHAR2(120)   | 4           | No             |
| CASH_EQUIV_FLAG   | The indicator of the cash equivalence.                                                                                                     | VARCHAR2(1)     | 5           | No             |

**ttldmdm.txt**

Business rules:

- This interface file contains tender type transaction information.
- This interface file cannot contain duplicate records for tndr\_type\_group\_idnt, tndr\_type\_id\_idnt, tran\_idnt, loc\_idnt, day\_dt, min\_idnt, rgstr\_idnt, and cshr\_idnt combination.
- This interface file follows the fact flat file interface layout standard.

| Name              | Description                                                                                                          | Data Type/Bytes | Field order | Required field |
|-------------------|----------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| TNDR_TYPE_ID_IDNT | The unique identifier for the tender type ID. An example of a tender type ID is Discover Card, Master Card, or Visa. | CHARACTER(6)    | 1           | Yes            |
| TRAN_IDNT         | The unique identifier of the transaction.                                                                            | VARCHAR2(30)    | 2           | Yes            |
| LOC_IDNT          | The unique identifier of the location.                                                                               | CHARACTER(10)   | 3           | Yes            |

| Name                | Description                                                                                                                               | Data Type/Bytes | Field order | Required field |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| DAY_DT              | The calendar day on which the transaction occurred.                                                                                       | DATE            | 4           | Yes            |
| MIN_IDNT            | The unique identifier of the minute.                                                                                                      | NUMBER(4)       | 5           | Yes            |
| RGSTR_IDNT          | The unique identifier of the register.                                                                                                    | CHARACTER(10)   | 6           | Yes            |
| CSHR_IDNT           | The unique identifier for a cashier.                                                                                                      | CHARACTER(10)   | 7           | Yes            |
| F_CC_SCAN_FLAG      | Indicates whether the credit card was scanned or manually entered. Valid values are 'Y' for scanned, or 'N' or Null for manually entered. | VARCHAR2(1)     | 8           | No             |
| F_TNDR_COUPON_COUNT | The total count of tender coupons used per transaction. Tender coupons are issues by the manufacturer as opposed to the store.            | NUMBER(16,4)    | 9           | No             |

| Name                  | Description                                                                                                                                        | Data Type/Bytes | Field order | Required field |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_TNDR_COUPON_AMT     | The total amount of tender coupons used per transaction. Tender coupons are issues by the manufacturer as opposed to the store.                    | NUMBER(18,4)    | 10          | No             |
| F_TNDR_COUPON_AMT_LCL | The total amount of tender coupons used per transaction, in local currency. Tender coupons are issued by the manufacturer as opposed to the store. | NUMBER(18,4)    | 11          | No             |
| F_TNDR_SLS_AMT        | The sales amount paid for with a particular tender type in primary currency.                                                                       | NUMBER(18,4)    | 12          | No             |
| F_TNDR_SLS_AMT_LCL    | The sales amount paid for with a particular tender type in local currency                                                                          | NUMBER(18,4)    | 13          | No             |

| Name                     | Description                                                                 | Data Type/Bytes | Field order | Required field |
|--------------------------|-----------------------------------------------------------------------------|-----------------|-------------|----------------|
| F_TNDR_RTRNS_SLS_AMT     | The return amount credited to a particular tender type in primary currency. | NUMBER(18,4)    | 14          | No             |
| F_TNDR_RTRNS_SLS_AMT_LCL | The return amount credited to a particular tender type in local currency.   | NUMBER(18,4)    | 15          | No             |

**ttltypdm.txt**

Business rules:

- This interface file contains user-defined totals.
- This interface file cannot contain duplicate records for a total\_type\_idnt.
- This interface file follows the dimension flat file interface layout standard.
- This interface file contains the complete snapshot of active information.

| Name            | Description                                             | Data Type/Bytes | Field order | Required field |
|-----------------|---------------------------------------------------------|-----------------|-------------|----------------|
| TOTAL_TYPE_IDNT | The original identifier for the total to be reconciled. | VARCHAR2(10)    | 1           | Yes            |
| TOTAL_TYPE_DESC | The description of the total type.                      | VARCHAR2(255)   | 2           | Yes            |

**vhreschddm.txt**

Business rules:

- This interface file contains the date and count of escheated vouchers. When a voucher escheats, the retailer releases all liability of the voucher to the state government. The quantity of escheated vouchers and the dates on which they are escheated are captured from this text file.
- This interface file cannot contain duplicate transactions for a day\_dt.
- This interface file follows the fact flat file interface layout standard.

| Name         | Description                                                                                                                                                        | Data Type/Bytes | Field order | Required field |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| DAY_DT       | The calendar day on which the transaction occurred.                                                                                                                | DATE            | 1           | Yes            |
| F_ESCH_COUNT | The total count of the escheated vouchers on a particular day.                                                                                                     | NUMBER(16,4)    | 2           | No             |
| F_ESCH_AMT   | The monetary amount of the escheated vouchers. If the voucher was never issued, the escheat amount is 0. If it was issued, the escheat amount is the issue amount. | NUMBER(18,4)    | 3           | No             |

**vchrmoveldsgdm.txt**

Business rules:

- This interface file contains issued and redeemed voucher information at the individual voucher level.
- This interface file cannot contain duplicate transactions for a vchr\_line\_no, vchr\_status\_cde combination.
- This interface file follows the fact flat file interface layout standard.

| Name            | Description                                                                                                                | Data Type/Bytes | Field order | Required field |
|-----------------|----------------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| VCHR_LINE_NO    | The unique identifier for an entry on this table. Corresponds to the unique identifier for a voucher in the source system. | VARCHAR2(20)    | 1           | Yes            |
| VCHR_STATUS_CDE | Indicates whether this is an issue (I) or redemption (R) record for this voucher.                                          | VARCHAR2(1)     | 2           | Yes            |
| LOC_IDNT        | The unique identifier of the location.                                                                                     | CHARACTER(10)   | 3           | Yes            |
| DAY_DT          | The calendar day on which the transaction occurred.                                                                        | DATE            | 4           | Yes            |

| Name              | Description                                                                                                          | Data Type/Bytes | Field order | Required field |
|-------------------|----------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| VCHR_AGE          | The age of the voucher in days.                                                                                      | NUMBER(6)       | 5           | Yes            |
| TNDR_TYPE_ID_IDNT | The unique identifier for the tender type ID. An example of a tender type ID is Discover Card, Master Card, or Visa. | CHARACTER(6)    | 6           | Yes            |
| RGSTR_IDNT        | The unique identifier of the register.                                                                               | CHARACTER(10)   | 7           | Yes            |
| CSHR_IDNT         | The unique identifier for a cashier.                                                                                 | CHARACTER(10)   | 8           | Yes            |
| F_AMT             | Amount for which this voucher was issued/redeemed in primary currency.                                               | NUMBER(18,4)    | 9           | No             |
| F_AMT_LCL         | Amount for which this voucher was issued/redeemed in the issue/redemption location's local currency.                 | NUMBER(18,4)    | 10          | No             |

**vchroutlwdm.txt**

Business rules:

- This interface file contains outstanding voucher information 'as of' the day\_dt. A voucher is outstanding if it is issued but not yet redeemed or escheated (that is, fully outstanding).
- This interface file cannot contain duplicate transactions for loc\_idnt, week, vchr\_age, tndr\_type\_id\_idnt, rgstr\_idnt, cshr\_idnt combination.
- This interface file follows the fact flat file interface layout standard.

| Name     | Description                                         | Data Type/Bytes | Field order | Required field |
|----------|-----------------------------------------------------|-----------------|-------------|----------------|
| LOC_IDNT | The unique identifier of the location.              | CHARACTER(10)   | 1           | Yes            |
| DAY_DT   | The calendar day on which the transaction occurred. | DATE            | 2           | Yes            |
| VCHR_AGE | The age of the voucher in days.                     | NUMBER(6)       | 3           | Yes            |

| Name              | Description                                                                                                          | Data Type/Bytes | Field order | Required field |
|-------------------|----------------------------------------------------------------------------------------------------------------------|-----------------|-------------|----------------|
| TNDR_TYPE_ID_IDNT | The unique identifier for the tender type ID. An example of a tender type ID is Discover Card, Master Card, or Visa. | CHARACTER(6)    | 4           | Yes            |
| RGSTR_IDNT        | The unique identifier of the register.                                                                               | CHARACTER(10)   | 5           | Yes            |
| CSHR_IDNT         | The unique identifier for a cashier.                                                                                 | CHARACTER(10)   | 6           | Yes            |
| F_OUT_COUNT       | The number of outstanding vouchers in this age band.                                                                 | NUMBER(16,4)    | 7           | No             |
| F_OUT_AMT         | The monetary amount of the outstanding vouchers, in primary currency.                                                | NUMBER(18,4)    | 8           | No             |
| F_OUT_AMT_LCL     | The monetary amount of the outstanding vouchers, in local currency.                                                  | NUMBER(18,4)    | 9           | No             |



## **Overview**

Oracle Retail Merchandising System (RMS) includes a convenient interface with your point-of-sale system (POS) that allows you to efficiently upload sales transaction data. RMS is able to accomplish these POS uploads because of the efficiency of its batch module that accepts your 'rolled-up' and formatted sales transaction data files into RMS. Once the data enters RMS, other modules take over the posting of that data to sales transaction, sales history, and stock-on-hand tables. This overview describes the upload and validation of sales transaction data from your POS to RMS and the relevant processes.

### **The POS upload process**

Before RMS can accept sales transaction data, you need to ensure that your data is correctly prepared. Then it is uploaded and processed into the TRAN\_DATA table.

#### **Preparing transaction data for upload**

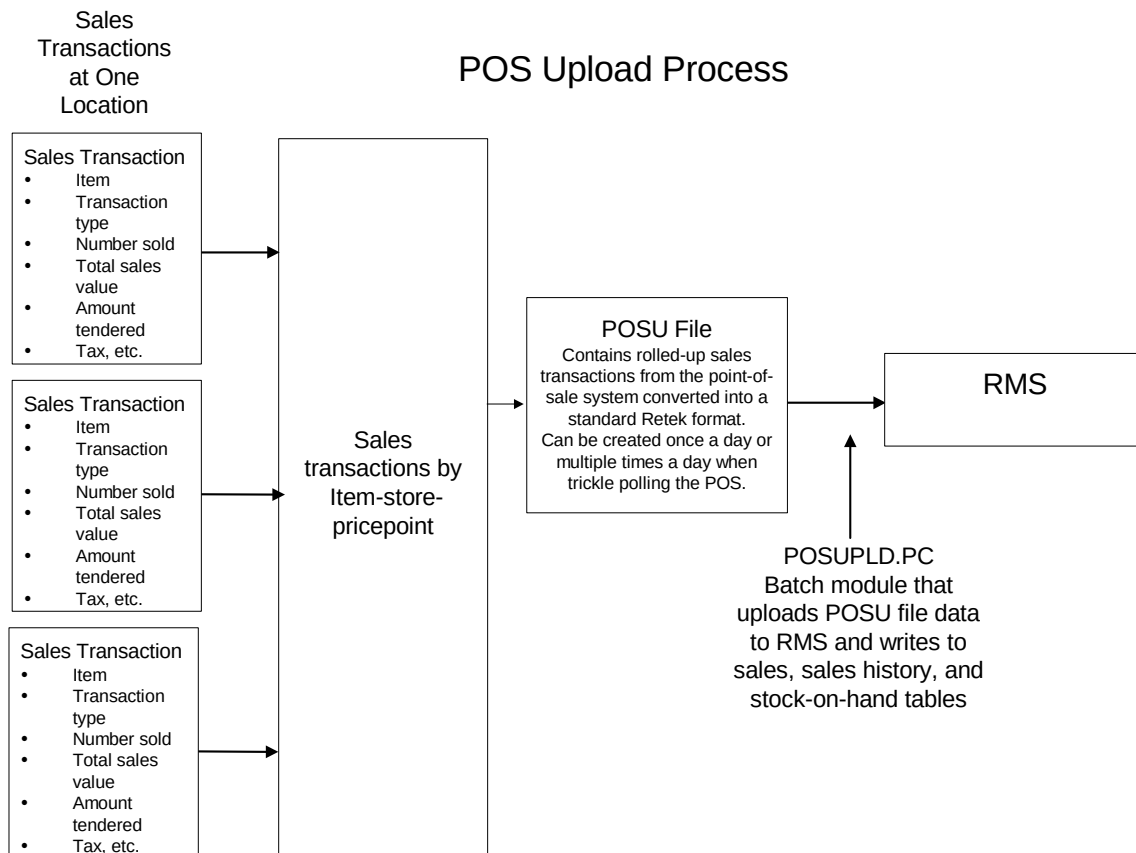
Two tasks need to be accomplished before transaction data is ready for upload to RMS. Initially, you roll up data in your transaction logs (called TLOGs), and then you convert the rolled-up files into a format that RMS can use.

Because you record transactions by the item sold, price, tax on the sale, amount tendered, and so on, you need to roll up these records to the item-day-store-price point level. The result of the rollup is that your transactions are described as the number of each item sold at a particular price at a store on one day.

After you roll up transaction data, the second task is to convert it to a file format that RMS can read, called the POSU file. POSUPLD.PC uploads and processes the POSU file into the TRAN\_DATA table.

#### **Upload transaction data**

POSUPLD.PC is RMS' batch program that uploads the POSU file into RMS. The module processes sales transaction data to various tables in RMS. The following diagram illustrates the upload process, which can occur once a day or multiple times during the day in a trickle-polling environment.



### Sales Transaction Data Uploaded To RMS

Note that the transaction type for a sale (as opposed to a return) can be a sale at a regular price, promotional price, or clearance price. Each sale price is considered a “price point”. The batch program POSUPLD.PC may run several times a day for each location.

**Note:** For descriptions of stock ledger implementation and accounting method options, see the chapter “Stock Ledger Batch” in this volume of the RMS Operations Guide.

### Processing POSU data

POSUPLD.PC accepts the POSU file as its input and processes its data. Processing is dependent upon such variables as Oracle Retail Sales Audit (ReSA) enabled, the stock ledger accounting method used (cost or retail), and value-added tax enabled.

POSUPLD.PC performs the following:

- Validates all item sales, unless the file is received from ReSA, where validation has already occurred.
- Converts the selling unit of measure (UOM) to the standard UOM (the stock ledger only holds standard UOM).
- Calculates value-added tax, where the retail accounting method is selected and the value-added tax indicator is enabled.
- Calculates the complete sales total (total retail, quantity, cost, and so on) for each item.

- Calculates promotional markdowns for use in writing transaction records for promotions.
- Processes pack sales by their individual component items.
- Posts transaction data records for sales and returns.
- Writes transactions for employee discounts and item wastage.
- Deals – POSUPLD.PC (Vendor funded promotions).
- Concession consignments.
- Catch-weight items (For more information, see the chapters “Transfers, Allocations, and RTV Batch” in this volume of the Operations Guide and “Item publication” in Volume 2 of the RMS Operations Guide.)
- Item transformation (POSUPLD.PC maps a sellable item to one or more orderable items – orderable items cannot be sold.)
- Item types
- Deposit items (For more information, see the chapter “Item publication” in this volume of the RMS Operations Guide.)
- SUB\_TRAN type for Sales Audit
- Calendar

Highlights of some of these processes follow, beginning in the next paragraph.

### Validate items

Unless POSUPLD.PC receives the POSU file from ReSA, it validates the sales or return transaction item’s number against the ITEM\_LOC table. Because the item can also be referenced by its item identifier, the module checks the reference item type on the ITEM\_MASTER table. Valid reference types are stored in the CODE\_DETAIL table under the code type of ‘UPCT’ as listed in the table that follows. After determining the reference type, the module locates the corresponding item number itself.

| RMS CODE TYPE | CODE   | CODE_DESC                 |
|---------------|--------|---------------------------|
| UPCT          | ITEM   | Retail Item Number        |
| UPCT          | UPC-A  | UPC-A                     |
| UPCT          | UPC-AS | UPC-A with Supplement     |
| UPCT          | UPC-E  | UPC-E                     |
| UPCT          | UPC-ES | UPC-E with Supplement     |
| UPCT          | EAN8   | EAN8                      |
| UPCT          | EAN13  | EAN13                     |
| UPCT          | EAN13S | EAN13 with Supplement     |
| UPCT          | ISBN   | ISBN                      |
| UPCT          | NDC    | NDC/NHRIC - National Drug |
| UPCT          | PLU    | PLU                       |
| UPCT          | VPLU   | Variable Weight PLU       |
| UPCT          | SSCC   | SSCC Shipper Carton       |

| RMS CODE TYPE | CODE  | CODE_DESC |
|---------------|-------|-----------|
| UPCT          | UCC14 | SCC-14    |

### Validate total amounts

Rolled-up sales transactions for individual items at the store are validated within POSUPLD.PC. Take a closer look at a list of sales transactions and how they are rolled up. Suppose that a store sells an item that is identified as Item Number 1234. During the day, sales for Item 1234 might look like this:

| Sales for Item Number 1234 (at one store during one day) |                      |                                     |                            |
|----------------------------------------------------------|----------------------|-------------------------------------|----------------------------|
| Transaction Number                                       | Number of Items Sold | Amount (in specified currency unit) | Price point (price reason) |
| 167                                                      | 1                    | 9.99                                | Regular                    |
| 395                                                      | 2                    | 18.00                               | Promotional                |
| 843                                                      | 1                    | 7.99                                | Clearance                  |
| 987                                                      | 3                    | 27.00                               | Promotional                |
| 1041                                                     | 1                    | 9.99                                | Regular                    |
| 1265                                                     | 4                    | 31.96                               | Clearance                  |

**Note:** The variation of the price per item in different transactions. This results from the price applied at the time of sale—the price point. Now look at the next table that shows the same transactions rolled up by item and price point.

| Number of Items Sold | Price Reason (price point) | Total Amount for Item-Price point (in currency) |
|----------------------|----------------------------|-------------------------------------------------|
| 2                    | Regular price              | 19.98                                           |
| 5                    | Promotional price          | 45.00                                           |
| 5                    | Clearance price            | 39.95                                           |

POSUPLD.PC takes the totals and looks for any discounts for transactions in the POSU file. It applies the discounts to an expected total dollar amount using the price listed for that item from the pricing table (PRICE\_HIST). It next compares this expected total against the reported total. If the program finds a discrepancy between the two amounts, it is reported. If the two totals match, the rollup is considered valid. If value-added tax (VAT) is included in any sales transaction amounts, it is removed from those transactions prior to the validation process.

## Post transaction data records

POSUPLD.PC posts transaction records to the TRAN\_DATA table primarily through its write\_tran\_data function. From the entire list of valid transaction codes (For the full list of transaction codes, see the chapter “General ledger batch” in this volume of the RMS Operations Guide), for the column TRAN\_CODE, POSUPLD.PC writes these codes:

| Transaction Code | Description                                                                                                                    |
|------------------|--------------------------------------------------------------------------------------------------------------------------------|
| 01               | Net Sales (retail & cost)                                                                                                      |
| 02               | Net sales (retail & cost) where - retail is always VAT exclusive, written only if system_options.stkldgr_vat_incl_retl_ind = Y |
| 03               | Non-inventory Items Sales/Returns                                                                                              |
| 04               | Customer Returns (retail & cost)                                                                                               |
| 05               | Non-inventory VAT Exclusive Sales                                                                                              |
| 06               | Deal Income (sales)                                                                                                            |
| 11               | Markup (retail only)                                                                                                           |
| 12               | Markup cancel (retail only)                                                                                                    |
| 13               | Permanent Markdown (retail only)                                                                                               |
| 14               | Markdown cancel (retail only)                                                                                                  |
| 15               | Promotional Markdown (retail only), including ‘in-store’ markdown                                                              |
| 20               | Purchases (retail & cost)                                                                                                      |
| 24               | Return to Vendor (RTV) from inventory (retail & cost)                                                                          |
| 60               | Employee discount (retail only)                                                                                                |

Note that where value-added-tax is enabled (system\_options table, stkldgr\_vat\_incl\_retl\_ind column shows ‘Y’) and the retail accounting method is also enabled, POSUPLD.PC writes an additional transaction record for code 02.

Note also that any items sold on consignment—where the department’s items are stocked as consignment, rather than normal (see the DEPS table, profit\_calc\_type column)—are written as a code 20 (Purchases) as well as a 01 (Net Sales) along with all other applicable transactions, like returns. The 20 reflects the fact that the item is purchased at the time it is sold, in other words, a consignment sale.

Sales transactions are written to sales, sales history, and stock-on-hand tables in RMS by POSUPLD.PC.

Additional batch modules in the batch process then begin to post that data to other RMS tables. The module HSTBLD.PC writes item-based transactions to historical sales data tables by subclass, class, and department. HSTBLD.PC runs daily after POSUPLD.PC.

## A Note about Oracle Retail Sales Audit and POSUPLD.PC

Oracle Retail offers customers an optional module called Oracle Retail Sales Audit (ReSA). Unlike the standard POS upload process to RMS that is described in this overview, ReSA accepts POS data at the transaction level for the store-day. The standard POS upload process described earlier requires the customer to roll up individual transactions to the item-store-day-price point level. ReSA validates individual sales transactions for a store day, offers a method to build and apply business audit rules, and lets users reconcile transaction errors, all prior to creating an output file rolled up to the department, class, and subclass level for posting to stock ledger tables by POSUPLD.PC.

## Batch Design Summary

The following batch designs are included in this functional area:

- HSTBLD.PC (History Build)
- HSTBLD\_DIFF.PC (Sales History Rollup by Diff IDs)
- HSTBLDMTH.PC (History Build Month)
- HSTBLDMTH\_DIFF.PC (Monthly Sales History Rollup by Diff IDs)
- HSTMTHUPD.PC (History Month Update)
- HSTPRG.PC (History Purge)
- HSTPRG\_DIFF.PC (Sales History Purge by Diff)
- HSTWKUPD.PC (History Week Update)
- POSUPLD.PC (Point of Sale Upload)

## hstbld (Sales History Rollup by Department, Class and Subclass)

### Functional Area

Sales Posting

### Module Affected

HSTBLD.PC

### Design Overview

The sales history rollup routine stores extract sales history information for each item from the ITEM\_MASTER, and ITEM\_LOC\_HIST (item location history) tables. The history information is rolled up to the subclass, class, and dept level to be written to:

dept\_sales\_hist (department/location/week/sales type), class\_sales\_hist (class/location/week/sales type), and subclass\_sales\_hist (subclass/location/week/sales type).

The rebuild program can be run in one of two ways:

First, if the program is run with a run-time parameter of 'rebuild', the program stores read data (dept, class, and subclass) off the manually input HIST\_REBUILD\_MASK table, which stores determine what to rebuild.

Secondly, if the program is run with a run-time parameter of 'weekly', the program stores build sales information for all dept/class/subclass combinations only for the current end of week date.

## Scheduling Constraints

| Schedule Information | Description                                                                                                                                                               |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle     | Phase 3 (weekly)<br>Phase ad-hoc (weekly)                                                                                                                                 |
| Scheduling Diagram   | Must run after complete weekly sales are updated by POSUPLD.PC. Also should be re-run on demand when a sales rollup request is given for a given dept, class or subclass. |
| Pre-processing       | N/A                                                                                                                                                                       |
| Post-Processing      | Run PREPOST.PC for HSTBLD to truncate the HIST_REBUILD_MASK table.                                                                                                        |
| Threading Scheme     | Threaded by location                                                                                                                                                      |

## Restart/Recovery

The logical unit of work for this program is set at the store/dept/class level. Threading is done by store using the v\_restart\_store view. The commit\_max\_ctr field on the RESTART\_CONTROL table stores determine the number of transactions that equal a logical unit of work.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table               | Select | Insert | Update | Delete |
|---------------------|--------|--------|--------|--------|
| DEPT_SALES_HIST     | No     | Yes    | Yes    | No     |
| CLASS_SALES_HIST    | No     | Yes    | Yes    | No     |
| SUBCLASS_SALES_HIST | Yes    | Yes    | Yes    | No     |
| ITEM_MASTER         | Yes    | No     | No     | No     |
| ITEM_LOC_HIST       | Yes    | No     | No     | No     |
| PERIOD              | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES    | Yes    | No     | No     | No     |
| HIST_REBUILD_MASK   | Yes    | No     | No     | No     |

## I/O Specification

N/A

## hstbld\_diff (Sales History Rollup by Diff\_IDs)

### Functional Area

Sales Posting.

### Module Affected

HSTBLD\_DIFF.PC

### Design Overview

The sales history rollup routine stores extract sales history information for each item\_parent from the ITEM\_LOC\_HIST table. The history information is rolled up to the item differentiator level to be written to: item\_diff\_loc\_hist and item\_parent\_loc\_hist.

For each item, data to be retrieved includes sales qty and stock. This data must be collected from several tables including ITEM\_LOC\_HIST, ITEM\_LOC, and ITEM\_MASTER.

### Scheduling Constraints

| Schedule Information | Description                                                                                                                                                                                                                          |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle     | Phase 3 (weekly)<br>AD-HOC (weekly)                                                                                                                                                                                                  |
| Scheduling Diagram   | Must run after complete weekly sales are updated by POSUPLD or after HSTBLD. Also should be re-run on demand when a sales rollup request is given for a given dept, class or subclass (run HSTBLD) or style/color (run this module). |
| Pre-processing       | N/A                                                                                                                                                                                                                                  |
| Post-Processing      | N/A                                                                                                                                                                                                                                  |
| Threading Scheme     | N/A                                                                                                                                                                                                                                  |

### Restart/Recovery

N/A

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| ITEM_PARENT_LOC_HIST | No     | Yes    | Yes    | No     |



| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| ITEM_DIFF_LOC_HIST | No     | Yes    | Yes    | No     |
| ITEM_LOC           | Yes    | No     | No     | No     |
| ITEM_LOC_HIST      | Yes    | No     | No     | No     |
| ITEM_MASTER        | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES   | Yes    | No     | No     | No     |
| PERIOD             | Yes    | No     | No     | No     |

### I/O Specification

N/A

## hstblmth (Monthly Sales History Rollup By Department, Class And Subclass)

### Functional Area

Sales posting

### Module Affected

HSTBLDMTH.PC

### Design Overview

The monthly sales history roll up routine stores extract sales history information for each item from the ITEM\_MASTER and ITEM\_LOC\_HIST\_MTH (item location history by month) tables. The history information is rolled up to the subclass, class and dept level to be written to: subclass\_sales\_hist\_mth (subclass/location/month/sales type), class\_sales\_hist\_mth (class/location/month/sales type) and dept\_sales\_hist\_mth (department/location/month/sales type).

This program may be run in parallel with HSTBLD since they both read from HIST\_REBUILD\_MASK. The table HIST\_REBUILD\_MASK table must not be truncated before both programs finish running.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                               |
|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 Ad Hoc (Monthly)                                                                                                                                                  |
| Scheduling Considerations | Must run after complete monthly sales are updated by POSUPLD. Also, should be re-run on demand when a sales rollup request is given for a given dept, class and subclass. |
| Pre-Processing            | N/A                                                                                                                                                                       |
| Post-Processing           | Run PREPOST.PC for HSTBLD to truncate the HIST_REBUILD_MASK table.                                                                                                        |

| Schedule Information | Description            |
|----------------------|------------------------|
| Threading Scheme     | Threaded by department |

**Restart/Recovery**

The logical unit of work for the hstbldmth module is department, location, sales type and end of month date with a recommended commit counter setting of 1,000. Processed records are committed each time the record counter equals the maximum recommended commit number.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| ITEM_MASTER             | Yes    | No     | No     | No     |
| ITEM_LOC_HIST_MTH       | Yes    | No     | No     | No     |
| SUBCLASS_SALES_HIST_MTH | Yes    | Yes    | No     | Yes    |
| CLASS_SALES_HIST_MTH    | Yes    | Yes    | No     | Yes    |
| DEPT_SALES_HIST_MTH     | No     | Yes    | No     | Yes    |
| HIST_REBUILD_MASK       | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES        | Yes    | No     | No     | No     |
| PERIOD                  | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**hstbldmth\_diff (Sales History Rollup By Diff Ids Per Month)****Functional Area**

Sales Posting.

**Module Affected**

HSTBLDMTH\_DIFF.PC

## Design Overview

The sales history rollup routine stores extract sales history information for each ITEM\_PARENT from the ITEM\_LOC\_HIST\_MTH table and rolls the data to month level. The history information is rolled up to the item differentiator level to be written to: item\_diff\_loc\_hist\_mth and item\_parentloc\_hist\_mth. For each item, data to be retrieved includes sales quantity and stock. This data must be collected from several tables including ITEM\_LOC\_HIST\_MTH, ITEM\_LOC, and ITEM\_MASTER.

## Scheduling Constraints

| Schedule Information      | Description                                                                  |
|---------------------------|------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (monthly)<br>AD-HOC (monthly)                                        |
| Scheduling Considerations | Must be run only at EOM date, before or after HSTBLD but before HSTBLD_POST. |
| Pre-Processing            | N/A                                                                          |
| Post-Processing           | N/A                                                                          |
| Threading Scheme          | N/A                                                                          |

## Restart/Recovery

N/A

## Locking Strategy

The package HSTBLD\_DIFF\_PROCESS locks the following tables for update:

ITEM\_DIFF\_LOC\_HIST\_MTH

ITEM\_PARENTLOC\_HIST\_MTH

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                   | Select | Insert | Update | Delete |
|-------------------------|--------|--------|--------|--------|
| ITEM_MASTER             | Yes    | No     | No     | No     |
| ITEM_LOC                | Yes    | No     | No     | No     |
| ITEM_LOC_HIST_MTH       | Yes    | No     | No     | No     |
| ITEM_DIFF_LOC_HIST_MTH  | No     | Yes    | Yes    | No     |
| ITEM_PARENTLOC_HIST_MTH | No     | Yes    | Yes    | No     |
| SYSTEM_VARIABLES        | Yes    | No     | No     | No     |

| Table  | Select | Insert | Update | Delete |
|--------|--------|--------|--------|--------|
| PERIOD | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**hstmthupd (Monthly Stock on Hand, Retail and Average Cost Values Update)****Functional Area**

Sales Posting

**Module Affected**

HSTMTHUPD.PC

**Design Overview**

This batch program runs monthly to update the stock on hand, retail values and average cost for each item/location on the ITEM\_LOC\_HIST\_MTH (item location history by month) table. If the item/location does not exist on the ITEM\_LOC\_HIST\_MTH table, then the new record is written to a comma delimited file which is later uploaded to ITEM\_LOC\_HIST\_MTH table using SQL\*Loader.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                                  |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (monthly)                                                                                                                                            |
| Scheduling Considerations | The program should be run on the last day of the month.                                                                                                      |
| Pre-Processing            | N/A                                                                                                                                                          |
| Post-Processing           | Run SQL*Loader using the control file hstmthupd.ctl to load data from the output file written by HSTMTHUPD.PC for non-existent records on ITEM_LOC_HIST_MTH. |
| Threading Scheme          | Threaded by location (store).                                                                                                                                |

**Restart/Recovery**

The logical unit of work for this program is the item/location record. Threading is done by store using the v\_restart\_store\_wh view. The commit\_max\_ctr field on the RESTART\_CONTROL table stores determine the number of transactions that equal a logical unit of work. Table-based restart/recovery is used.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| SYSTEM_VARIABLES  | Yes    | No     | No     | No     |
| PERIOD            | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS    | Yes    | No     | No     | No     |
| ITEM_LOC          | Yes    | No     | No     | No     |
| ITEM_LOC_SOH      | Yes    | No     | No     | No     |
| ITEM_LOC_HIST_MTH | Yes    | No     | Yes    | No     |

**I/O Specification**

This program stores create a comma delimited output data file (load\_hstmthupd.dat or user-supplied name) for SQL\*Loader to upload the data to the ITEM\_LOC\_HIST\_MTH table. The control script for the SQL \* Loader is HSTMTHUPD.CTL.

**hstprg (Purge Sales History)****Functional Area**

Sales Posting.

**Module Affected**

HSTPRG.PC

**Design Overview**

Deletes records from ITEM\_LOC\_HIST, SUBCLASS\_SALES\_HIST, CLASS\_SALES\_HIST, DEPT\_SALES\_HIST and DAILY\_SALES\_DISCOUNT tables, where data is older than the specified number of months. Number of months for retention of fashion style history is specified by unit\_options.ITEM\_HISTORY\_MONTHS.

**Scheduling Constraints**

| Schedule Information      | Description      |
|---------------------------|------------------|
| Processing Cycle          | AD-HOC (monthly) |
| Scheduling Considerations | N/A              |
| Pre-Processing            | N/A              |
| Post-Processing           | N/A              |
| Threading Scheme          | N/A              |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| UNIT_OPTIONS         | Yes    | No     | No     | No     |
| PERIOD               | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS       | Yes    | No     | No     | No     |
| ITEM_LOC_HIST        | No     | No     | No     | Yes    |
| SUBCLASS_SALES_HIST  | No     | No     | No     | Yes    |
| CLASS_SALES_HIST     | No     | No     | No     | Yes    |
| DEPT_SALES_HIST      | No     | No     | No     | Yes    |
| DAILY_SALES_DISCOUNT | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**hstprg\_diff (Sales History Purge by Diff)****Functional Area**

Sales posting

**Module Affected**

HSTPRG\_DIFF.PC

**Design Overview**

The tables, ITEM\_DIFF\_LOC\_HIST and ITEM\_PARENT\_LOC\_HIST are purged of sales history differentiator data, which is older than a specified system set date. This date is stored in the item\_history\_months column of UNIT\_OPTIONS.

**Scheduling Constraints**

| Schedule Information      | Description                                                                 |
|---------------------------|-----------------------------------------------------------------------------|
| Processing Cycle          | Ad Hoc (weekly)                                                             |
| Scheduling Considerations | Must run after HSTBLD_DIFF.PC. Also can be re-run on demand when requested. |

| Schedule Information | Description |
|----------------------|-------------|
| Pre-Processing       | N/A         |
| Post-Processing      | N/A         |
| Threading Scheme     | N/A         |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| UNIT_OPTIONS         | Yes    | No     | No     | No     |
| PERIOD               | Yes    | No     | No     | No     |
| ITEM_DIFF_LOC_HIST   | No     | No     | No     | Yes    |
| ITEM_PARENT_LOC_HIST | No     | No     | No     | Yes    |

**I/O Specification**

N/A

## hstwkupd (Weekly Stock on Hand and Retail Value Update for Fashion Item/Location)

**Functional Area**

Sales Posting.

**Module Affected**

HSTWKUPD.PC

**Design Overview**

This program runs weekly to update the current stock on hand, retail values and average cost for each item/location on ITEM\_LOC\_HIST. The program must be run on the last day of the week as scheduled.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                            |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (weekly)                                                                                                                                       |
| Scheduling Considerations | N/A                                                                                                                                                    |
| Pre-Processing            | N/A                                                                                                                                                    |
| Post-Processing           | Run SQL*Loader using the control file hstwkupd.ctl to load data from the output file written by HSTWKUPD.PC for non-existent records on ITEM_LOC_HIST. |
| Threading Scheme          | Thread by location                                                                                                                                     |

### Restart/Recovery

The logical unit of work for HSTWKUPD is item/location. The program is threaded by location using the v\_restart\_store\_wh view.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| ITEM_LOC           | Yes    | No     | No     | No     |
| ITEM_LOC_SOH       | Yes    | No     | No     | No     |
| V_RESTART_STORE_WH | Yes    | No     | No     | No     |
| ITEM_LOC_HIST      | Yes    | No     | Yes    | No     |
| SYSTEM_VARIABLES   | Yes    | No     | No     | No     |
| PERIOD             | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS     | Yes    | No     | No     | No     |

### I/O Specification

This program stores create a comma delimited output data file (load\_hstwkupd.dat or user-supplied name) for SQL\*Loader to upload data to ITEM\_LOC\_HIST table. The control script for the SQL \* Loader is HSTWKUPD.CTL.



## posupld (Point Of Sales Upload)

### Functional Area

Sales Posting

### Module Affected

POSUPLD.PC

### Design Overview

The purpose of the POSUPLD.PC module is to process sales and return details from an external point of sale system. The sales/return transactions are validated against Oracle Retail item/store relations to ensure the sale is valid, but this validation process can be eliminated if the sales that are being passed in, is screened by sales auditing (ReSA). The following common functions are performed on each sales/return record read from the input file:

- Read sales/return transaction record
- Lock associated record in RMS
- Validate item sale
- Check whether VAT maintenance is required, and if so determine the VAT amount for the sale
- Write all financial transactions for the sale and any relevant markdowns to the stock ledger.
- Post item/location/week sales to the relevant sales history tables
- Perform last sales processing to maintain accurate sales information in the system

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                                                                                                                          |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 2 (Daily)                                                                                                                                                                                                                                                                      |
| Scheduling Considerations | This program is run daily in Phase 2 of RMS' batch cycle as point-of-sales data, in the form of the POSU file, becomes available. It can be run multiple times a day in a trickle-polling environment.<br>It should be run after SAEXPRMS.PC when Oracle Retail Sales Audit is used. |
| Pre-Processing            | N/A                                                                                                                                                                                                                                                                                  |
| Post-Processing           | Post-processing (prepost posupld post) stores update invoice matching tables.                                                                                                                                                                                                        |
| Threading Scheme          | N/A                                                                                                                                                                                                                                                                                  |

### Restart/Recovery

The logical unit of work for POSUPLD is a valid item sales transaction at a given store location. The logical unit of work is defined as a group of these transaction records. Oracle Retail standard file-based restart/recovery logic is used. Records are committed to

the database when the maximum commit counter is reached. The recommended commit counter setting is 10000 records (subject to change based on experimentation).

### Locking Strategy

Because POSUPLD can be run multiple times a day in a trickle-polling system, a locking mechanism is put in place to allow on-line transactions and POSUPLD to run at the same time. The following tables are locked for update:

- VAT\_HISTORY
- DAILY\_SALES\_DISCOUNT
- ITEM\_LOC\_SOH
- ITEM\_LOC\_HIST
- ITEM\_LOC\_HIST\_MTH
- EDI\_DAILY\_SALES
- INVC\_MERCH\_VAT
- DEAL\_ACTUALS\_ITEM\_LOC

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| VAT_HISTORY           | No     | Yes    | Yes    | No     |
| DAILY_SALES_DISCOUNT  | No     | Yes    | Yes    | No     |
| LOAD_ERR              | No     | Yes    | No     | No     |
| STORE                 | Yes    | No     | No     | No     |
| CURRENCIES            | Yes    | No     | No     | No     |
| CLASS                 | Yes    | No     | No     | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| DEPS                  | Yes    | No     | No     | No     |
| RPM_PROMO             | Yes    | No     | No     | No     |
| RPM_PROMO_COMP        | Yes    | No     | No     | No     |
| DEAL_HEAD             | Yes    | No     | No     | No     |
| DEAL_COMP_PROM        | Yes    | No     | No     | No     |
| DEAL_ACTUALS_FORECAST | Yes    | No     | No     | No     |
| ITEM_LOC              | Yes    | No     | No     | No     |
| ITEM_LOC_SOH          | Yes    | No     | Yes    | No     |

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| VAT_ITEM              | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| ITEM_SUPPLIER         | Yes    | No     | No     | No     |
| SUPS                  | Yes    | No     | No     | No     |
| TERMS                 | Yes    | No     | No     | No     |
| PRICE_HIST            | Yes    | No     | No     | No     |
| TEMP_TRAN_DATA        | No     | Yes    | No     | No     |
| ITEM_LOC_HIST         | Yes    | Yes    | Yes    | No     |
| ITEM_LOC_HIST_MTH     | Yes    | Yes    | Yes    | No     |
| EDI_DAILY_SALES       | Yes    | Yes    | Yes    | No     |
| ORDHEAD               | Yes    | Yes    | No     | No     |
| INVC_HEAD             | Yes    | Yes    | No     | No     |
| INVC_MERCH_VAT        | Yes    | Yes    | Yes    | No     |
| INVC_XREF             | No     | Yes    | No     | No     |
| INVC_DETAIL_TEMP2     | No     | Yes    | No     | No     |
| INVC_DETAIL           | Yes    | No     | No     | No     |
| CODE_DETAIL           | Yes    | No     | No     | No     |
| UOM_CLASS             | Yes    | Yes    | No     | No     |
| ITEM_XFORM_HEAD       | Yes    | No     | No     | No     |
| ITEM_XFORM_DETAIL     | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_LOC | Yes    | No     | No     | No     |
| TRAN_DATA             | No     | Yes    | No     | No     |
| INVC_DETAIL_TEMP      | No     | Yes    | No     | No     |
| INVC_HEAD_TEMP        | No     | Yes    | No     | No     |
| CONCESSION_DATA       | No     | Yes    | No     | No     |
| DEAL_ACTUALS_ITEM_LOC | Yes    | Yes    | Yes    | No     |
| V_PACKSKU_QTY         | Yes    | No     | No     | No     |
| IF_ERRORS             | No     | Yes    | No     | No     |

## I/O Specification

Input File Layout

| Record Name        | Field Name                  | Field Type | Default Value                | Description                                                                                                              |
|--------------------|-----------------------------|------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| File Header        | File Type Record Descriptor | Char(5)    | FHEAD                        | Identifies file record type                                                                                              |
|                    | File Line Identifier        | Number(10) | Specified by external system | ID of current line being processed by input file.                                                                        |
|                    | File Type Definition        | Char(4)    | POSU                         | Identifies file as 'POS Upload'                                                                                          |
|                    | File Create Date            | Char(14)   |                              | Date file was written by external system                                                                                 |
|                    | Location Number             | Number(10) |                              | Store identifier                                                                                                         |
|                    | Vat include indicator       | Char(1)    |                              | Determines whether or not the store stores values including vat. Not required but populated by Oracle Retail sales audit |
|                    | Vat region                  | Number(4)  |                              | Vat region the given location is in. Not required but populated by Oracle Retail Sales Audit.                            |
|                    | Currency code               | Char(3)    |                              | Currency of the given location. Not required but populated by Oracle Retail sales audit                                  |
|                    | Currency retail decimals    | Number(1)  |                              | Number of decimals supported by given currency for retails. Not required but populated by Oracle Retail sales audit      |
| Transaction Header | File Type Record Descriptor | Char(5)    | THEAD                        | Identifies transaction record type                                                                                       |
|                    | File Line Identifier        | Number(10) | Specified by external system | ID of current line being processed by input file.                                                                        |
|                    | Transaction Date            | Char(14)   | Transaction date             | Date sale/return transaction was processed at the POS                                                                    |

| Record Name | Field Name      | Field Type | Default Value | Description                                                                                                                                           |
|-------------|-----------------|------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Item Type       | Char(3)    | REF or ITM    | Item type is represented as a REF or ITM                                                                                                              |
|             | Item Value      | Char(25)   |               | The ID number of an ITM or REF                                                                                                                        |
|             | Dept            | Number(4)  |               | Dept of item sold or returned. Not required but populated by Oracle Retail Sales Audit.                                                               |
|             | Class           | Number(4)  |               | Class of item sold or returned. Not required but populated by Oracle Retail Sales Audit                                                               |
|             | Subclass        | Number(4)  |               | Subclass of item sold or returned. Not required but populated by Oracle Retail Sales Audit.                                                           |
|             | Pack Indicator  | Char(1)    |               | Pack indicator of item sold or returned. Not required but populated by Oracle Retail Sales Audit.                                                     |
|             | Item level      | Number(1)  |               | Item level of item sold or returned. Not required but populated by Oracle Retail Sales Audit.                                                         |
|             | Tran level      | Number(1)  |               | Tran level of item sold or returned. Not required but populated by Oracle Retail Sales Audit.                                                         |
|             | Wastage Type    | Char(6)    |               | Wastage type of item sold or returned. Not required but populated by Oracle Retail Sales Audit.                                                       |
|             | Wastage Percent | Number(12) |               | Wastage Percent*10000 (4 implied decimal places.), wastage percent of item sold or returned. Not required but populated by Oracle Retail Sales Audit. |

| Record Name | Field Name              | Field Type | Default Value                    | Description                                                                                                                                                            |
|-------------|-------------------------|------------|----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             | Transaction Type        | Char(1)    | 'S' – sales<br>'R' - return      | Transaction type code to specify whether transaction is a sale or a return.                                                                                            |
|             | Drop Shipment Indicator | Char(1)    | 'Y'<br>'N'                       | Indicates whether the transaction is a drop shipment or not. If it is a drop shipment, indicator is 'Y'. This field is not required, but is defaulted to 'N' if blank. |
|             | Total Sales Quantity    | Number(12) |                                  | Total sales quantity * 10000 (4 implied decimal places), number of units sold at a particular location.                                                                |
|             | Selling UOM             | Char(4)    |                                  | UOM at which this item was sold.                                                                                                                                       |
|             | Sales Sign              | Char(1)    | 'P' - positive<br>'N' - negative | Determines if the Total Sales Quantity and Total Sales Value are positive or negative.                                                                                 |
|             | Total Sales Value       | Number(20) |                                  | Total Sales Value * 10000 (4 implied decimal places), sales value, net sales value of goods sold.                                                                      |
|             | Last Modified Date      | Char(14)   |                                  | For VBO future use.                                                                                                                                                    |
|             | Catchweight Indicator   | Char(1)    | NULL                             | Indicates if the item is a catch weight item. Valid values are 'Y' or NULL.                                                                                            |
|             | Actual Weight Quantity  | Number(12) | NULL                             | Actual Weight Quantity*10000 (4 implied decimal places), the actual weight of the item, only populated if catchweight_ind = 'Y'.                                       |
|             | Sub Trantype Indicator  | Char(1)    | NULL                             | Tran type for ReSA<br>Valid values are 'A', 'D', NULL.                                                                                                                 |

| Record Name         | Field Name                  | Field Type | Default Value                | Description                                                                                     |
|---------------------|-----------------------------|------------|------------------------------|-------------------------------------------------------------------------------------------------|
| Transaction Detail  | File Type Record Descriptor | Char(5)    | TDETL                        | Identifies transaction record type.                                                             |
|                     | File Line Identifier        | Number(10) | Specified by external system | ID of current line being processed by input file.                                               |
|                     | Promotional Tran Type       | Char(6)    |                              | Code for promotional type from code_detail, code_type = 'PRMT'                                  |
|                     | Promotion Number            | Number(10) |                              | Promotion number from the RMS.                                                                  |
|                     | Sales Quantity              | Number(12) |                              | Sales quantity*10000 (4 implied decimal places.), number of units sold in this prom type.       |
|                     | Sales Value                 | Number(20) |                              | Sales value*10000 (4 implied decimal places.), value of units sold in this prom type.           |
|                     | Discount Value              | Number(20) |                              | Discount quantity*10000 (4 implied decimal places.), value of discount given in this prom type. |
|                     | Promotion Component         | Number(10) |                              | Links the promotion to additional pricing attributes                                            |
| Transaction Trailer | File Type Record Descriptor | Char(5)    | TTAIL                        | Identifies file record type                                                                     |
|                     | File Line Identifier        | Number(10) | Specified by external system | ID of current line being processed by input file.                                               |
|                     | Transaction Count           | Number(6)  | Specified by external system | Number of TDETL records in this transaction set                                                 |
| File Trailer        | File Type Record Descriptor | Char(5)    | FTAIL                        | Identifies file record type                                                                     |
|                     | File Line Identifier        | Number(10) | Specified by external system | ID of current line being processed by input file.                                               |

| Record Name | Field Name          | Field Type | Default Value | Description                                                                                   |
|-------------|---------------------|------------|---------------|-----------------------------------------------------------------------------------------------|
|             | File Record Counter | Number(10) |               | Number of records/transactions processed in current file (only records between fhead & ftail) |

### Output Files

#### Invalid Item/Store File:

The Invalid Item/Store File stores only be written when a transaction holds an item that does not exist at the processed location. In the event this happens, the relationship is created during the program execution and processing stores continue with the item and store number being written to this file for reporting.

#### VAT File:

The VAT file stores only be written if a particular item cannot retrieve a VAT rate when one is expected (for example, the `system_options.vat_ind` is on). In this event, a non-fatal error stores occur against the transaction and a record is written to this file and the reject file.

#### Reject File:

The reject file should be able to be re-processed directly. The file format stores therefore be identical to the input file layout. A reject line counter is kept in the program and is required to ensure that the file line count in the trailer record matches the number of rejected records. A reject file is created in all cases. If no errors occur, the reject file stores consist only of a file header and trailer record.

#### Lock Reject File:

Lock reject file is written when TDETL record's validation has failed and item in that record is under a lock.



## Scheduled Item Maintenance Batch

### Overview

Scheduled item maintenance functionality allows you to assign items to item lists and to associate item lists with location lists, for both store and warehouse locations. In addition, there is a security feature that limits editing a list to the user who has created it.

### Security Feature for Item Lists

RMS has a security feature that limits the editing of a list to the user who created it. The item list table SKULIST\_HEAD and the location list table LOC\_LIST\_HEAD both contain a user\_security\_ind column. If the value in this column for a row is Y (Yes), this means that security is enabled. In this case, an Oracle package compares the CREATE\_ID to the logged on user. If there is a match, that person can modify the record. Otherwise, the user cannot modify the record. If the value in this column for a row is N (No), this means that security is not enabled, and the logged on user can modify the record.

### Batch Design Summary

The following batch design is included in this functional area:

- SITMAIN.PC (Scheduled Item Maintenance Main)

### sitmain (Scheduled Item Maintenance Main)

#### Functional Area

Item Maintenance

#### Module Affected

SITMAIN.PC

#### Design Overview

This module performs item/location-related updates. The processing is two-fold.

The first process involves tracking additions to item lists or locations lists. The program retrieves the latest status of an item/location if there is a pre-existing link that involves additions to an item or location list. If the given item/location relationship already exists, then the status and status\_update\_date field on the ITEM\_LOC table is updated accordingly. A new item/loc relationship is created if the relationship does not yet exist. All records in the SIT\_EXPLODE table with an update\_ind of 'Y' is updated to 'N' after all update/inserts into ITEM\_LOC are completed.

Scheduled item maintenance is performed in the second process. Updates are done to the status on the ITEM\_LOC table as specified in the SIT\_DETAIL table for the given item list or location list. If the item/location relationship does not exist at the time of update, it is created. Records are deleted from the SIT\_DETAIL table after processing.

### Scheduling Constraints

| Schedule Information      | Description                              |
|---------------------------|------------------------------------------|
| Processing Cycle          | AD HOC (daily)                           |
| Scheduling Considerations | This module should run after LCLRBLD.PC. |
| Pre-Processing            | N/A                                      |
| Post-Processing           | N/A                                      |
| Threading Scheme          | N/A                                      |

### Restart/Recovery

This program has inherent restart ability because records are deleted from SIT\_DETAIL as they are processed. The logical unit of work is an item/location combination.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table         | Select | Insert | Update | Delete |
|---------------|--------|--------|--------|--------|
| SIT_EXPLODE   | Yes    | No     | Yes    | No     |
| SIT_DETAIL    | Yes    | No     | No     | Yes    |
| ITEM_LOC      | Yes    | Yes    | Yes    | No     |
| MC_REJECTIONS | No     | Yes    | No     | No     |
| ITEM_MASTER   | Yes    | No     | No     | No     |
| POS_MODS      | No     | Yes    | No     | No     |
| PRICE_HIST    | No     | Yes    | No     | No     |
| ITEM_LOC_SOH  | No     | Yes    | No     | No     |

### I/O Specification

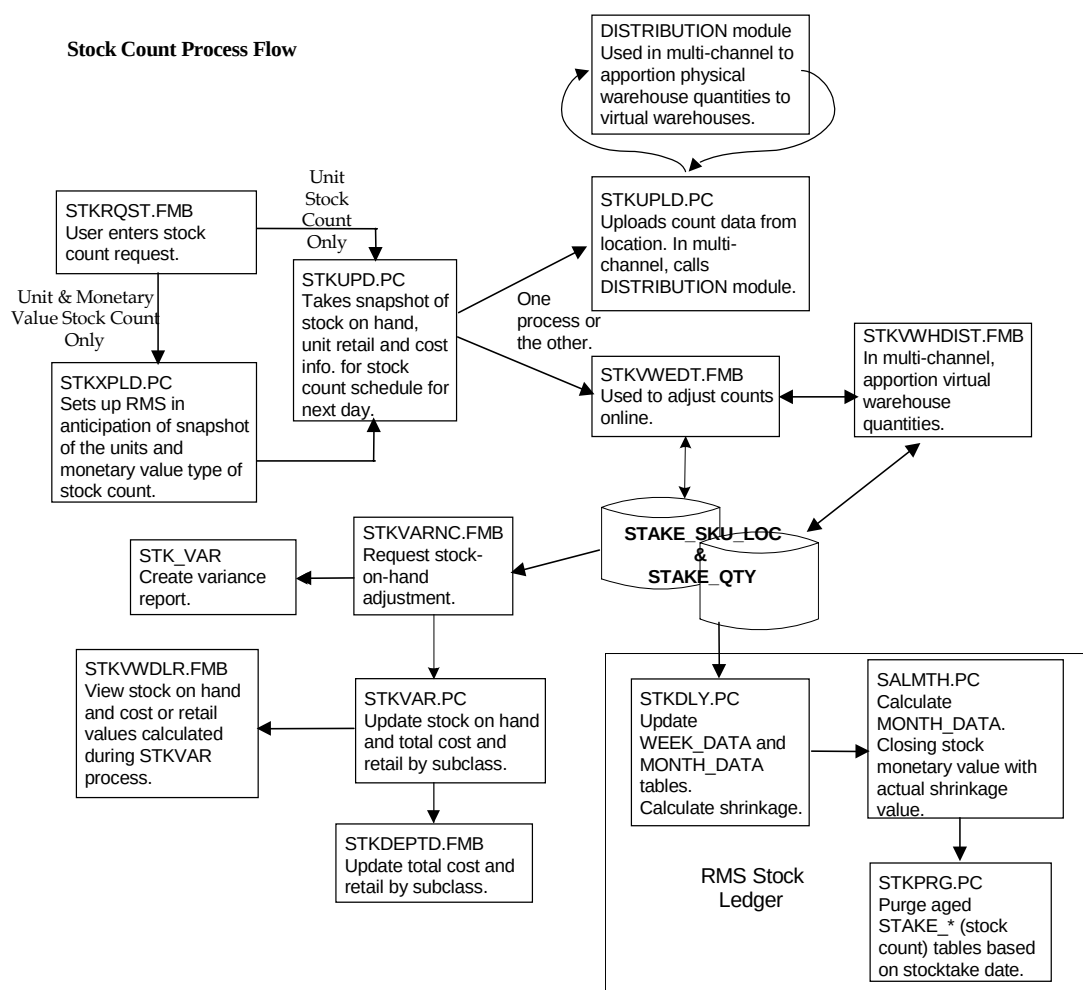
N/A

## Stock Count Batch

### Overview

A stock count is a comparison of an inventory snapshot at a point in time to an actual inventory count received from a location. From a stocktake request form, the RMS user can tell RMS to perform a stock count for unit counts of an item list, or a stock count of units for items at one location along with their monetary values. A location in a single channel environment is any stockholding location, meaning any store or warehouse. A location in a multi-channel environment is any stockholding store or non-stockholding warehouse, meaning a physical warehouse. Any differences between the snapshot and the actual physical count processed to RMS from the location are viewable in a variance report. Stock-on-hand adjustments can then be made in order to match the booked stock on hand to actual physical counts. Finally, if the user chooses a unit and monetary value stock count, the resulting data can be used to update the stock ledger. This overview focuses on those batch programs that set up and process stock count data, stock on hand adjustments, and stock ledger updates.

**Stock Count Process Flow**



**Stock Count Process (Including Multi-Channel Processes)**

## Stock Count Types: Units Versus Units and Monetary Values

A stock count can include a count of item units only or a count of item units along with their monetary value. Here are the differences between the two.

### Stock Count – Unit Only

An item list is selected when requesting a unit-only stock count so that items can be grouped together as required. As a result of this type of stock count the:

- Stock on hand is adjusted to reflect the physical count
- Stock Ledger is not adjusted
- Shrinkage monetary values and percent are not calculated

### Stock Count – Unit and Monetary Value

The department, class, or subclass is used when requesting a stock count of units and monetary value because the results of the count are moved into the stock ledger, which is maintained at the subclass level. As a result of this type of stock count the:

- Stock-on-hand is adjusted to reflect the physical count units
- Stock ledger is adjusted by physical count dollars
- Shrinkage monetary value and percent is calculated at the subclass-location level

## Stock Count Process

The steps in the stock count process follow the path described in this section. Detailed descriptions of each batch module are in the following section.

1. User requests a stock count from the STKRQST form.
2. Batch module STKXPLD.PC sets up RMS tables in anticipation of the snapshot of the units and monetary value type of stock count.
3. Batch module STKUPD.PC takes the snapshot of the stock on hand, retail, and cost (both unit cost and average cost) information for stock counts scheduled for the next day.
4. Batch module STKUPLD.PC processes count data for the selected location that are contained in the INV\_BAL file previously translated by the LIFSTKUP.PC module. If the location is a physical warehouse in a multi-channel environment, STKUPLD.PC calls the distribution module to apportion the data to the virtual warehouses associated with that physical warehouse.
5. After STKUPLD.PC runs, the counts for a physical location can be adjusted online from the STKVWEDT form. For multi-channel the quantities among virtual warehouses in a physical warehouse can be adjusted from the STKVWHDIST form.
6. The results of the actual count can be compared online to the previously taken snapshot of book stock. Variances between the book stock and the physical count can be reconciled through the STKVARNC form.

---

**Note:** At this point in the process, STAKE\_SKU\_LOC contains both snapshot and actual adjusted count data. The user can now review any variances between snapshot and actual count for item-location combinations. Online variance review is done online through the STKVARNC form where a stock-on-hand adjustment can be requested.

---

7. The batch module STKVAR.PC then updates stock on hand and the total cost or retail values for subclasses for the stock ledger.

---

**Note:** If the stock count has included monetary values in addition to item-location counts, the last steps are to correct the book stock value on the stock ledger and to calculate shrinkage rates.

---

8. The final step in the processing of unit and monetary value counts is handled by the STKDLY.PC module that updates the stock ledger and calculates inventory shrinkage.
9. One last batch module, STKPRG.PC, purges dated stock count tables.

## Stock Count Request

Stock counts result from a user request or by the creation of a stock count schedule. This overview focuses on the user request method. The request begins whenever a user opens the STKRQST form in RMS. This form asks the user to:

- Select a stock count of units only, or units and monetary value.
  - If a unit-only count is desired, the user enters an item list.
  - If a unit and monetary value count is desired, the user must select a department, class, or subclass. This allows RMS to update the stock ledger, which is maintained at the subclass level.
- Select a date on which the snapshot is to occur.
- Select locations for the stock count.

## Batch Design Summary

The following batch designs are included in this functional area:

- LIFSTKUP (Stock Upload Conversion)
- STKDLY.PC (Stock Count Shrinkage Update)
- STKPRG.PC (Purge Stock Count)
- STKSCHEDXPLD.PC (Scheduled Stock Count Explode)
- STKUPD.PC (Stock Count Snapshot)
- STKUPLD.PC (Upload Stock Count)
- STKVAR.PC (Stock Count on Hand Updates)
- STKXPLD.PC (Stock Count Explode)

## lifstkup (Stock Upload Conversion)

### Functional Area

Stock count

## Module Affected

LIFSTKUP.PC

## Design Overview

This program converts a distribution management inventory balance upload file into the Oracle Retail standard flat file for STKUPLD.PC to process.

This program verifies that the inventory data is for the requested cycle count and warehouse before proceeding. Other data needed for the RMS flat file is obtained from RMS tables and inserted in the RMS flat file.

## Scheduling Constraints

| Schedule Information      | Description                                                                                               |
|---------------------------|-----------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 1 (daily)                                                                                           |
| Scheduling Considerations | This program should run before STKUPLD.PC and after the warehouse management's inv_bal_upload.sh program. |
| Pre-Processing            | N/A                                                                                                       |
| Post-Processing           | N/A                                                                                                       |
| Threading Scheme          | N/A – File-based processing                                                                               |

## Restart/Recovery

Oracle Retail standard file-based restart/recovery is used. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 1000 records (subject to change based on implementation).

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| WH             | Yes    | No     | No     | No     |
| ITEM_MASTER    | Yes    | No     | No     | No     |
| STAKE_HEAD     | Yes    | No     | No     | No     |
| STAKE_LOCATION | Yes    | No     | No     | No     |

## I/O Specification

### Input format

|                    |                                                                                      |                                                                                                                                                                                 |
|--------------------|--------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DC_DEST_ID         | 11 - Number(10) + 1 for trailing space.                                              | Unique identifier for the DC                                                                                                                                                    |
| TRANSACTION_DATE   | 15 - Date(14) + 1 for trailing space.                                                | Date of the run                                                                                                                                                                 |
| ITEM_ID            | 26 - Varchar2(25) + 1 for trailing space.                                            | Uniquely identifies the item.                                                                                                                                                   |
| AVAILABLE_QTY      | 15 - Number(12) + 1 for leading sign and + 1 for decimal and + 1 for trailing space. | Units available for distribution                                                                                                                                                |
| DISTRIBUTED_QTY    | 14 Number(12) + 1 for decimal and + 1 for trailing space.                            | Units distributed include: Units distributed but not yet picked, units picked but not yet manifested, units manifested but not yet shipped.                                     |
| RECEIVED_QTY       | 15 Number(12) + 1 for leading sign and + 1 for decimal and + 1 for trailing space.   | Units received but not put away.                                                                                                                                                |
| TOTAL_QTY          | 14 Number(12,4) + 1 for decimal and + 1 for trailing space.                          | Sum of all units that physically exist: container status of: I, D, M, R, T, X.                                                                                                  |
| AVAILABLE_WEIGHT   | 15 - Number(12,4) + 1 for leading sign + 1 for decimal + 1 for trailing space.       | Weight available for distribution of catch weight items.                                                                                                                        |
| RECEIVED_WEIGHT    | 14 - Number(12,4) + 1 for decimal + 1 for trailing space.                            | Weight received but not put away for catch weight items.                                                                                                                        |
| DISTRIBUTED_WEIGHT | 14 - Number(12,4) + 1 for decimal + 1 for trailing space.                            | Weight distributed includes: weight distributed but not yet picked, weight picked but not yet manifested, weight manifested but not yet shipped (value only catch weight items) |
| TOTAL_WEIGHT       | 13 - Number(12,4) + 1 for decimal.                                                   | Sum of all weight that physically exist: container status of: I, D, M, R, T, X. For catch weight items.                                                                         |

### Output File Layout

| Record Name | Field Name                  | Field Type                   | Default Value | Description                                                |
|-------------|-----------------------------|------------------------------|---------------|------------------------------------------------------------|
| FHEAD       | file type record descriptor | Char(5)                      | FHEAD         | hardcode 'FHEAD'                                           |
|             | file line identifier        | Number(10)                   | 0000000001    | Id of current line being processed., hardcode '000000001'  |
|             | file type                   | Char(4)                      | 'STKU'        | hardcode 'STKU'                                            |
|             | stocktake_date              | Date(14)<br>YYYYMMDDHH24MISS |               | stake_head.stocktake_date                                  |
|             | file create date            | Date(14)<br>YYYYMMDDHH24MISS |               | date written by convert program                            |
|             | cycle count                 | Number(8)                    |               | stake_head.cycle_count                                     |
|             | Location type               | Char(1)                      | 'W'           | Hardcode 'W'                                               |
|             | location                    | Number(10)                   |               | stake_location.wh                                          |
| FDETL       | file type record descriptor | Char(5)                      | FDETL         | hardcode 'FDETL'                                           |
|             | file line identifier        | Number(10)                   |               | Id of current line being processed, internally incremented |
|             | Item type                   | Char(3)                      | 'ITM'         | Hardcode 'ITM'                                             |
|             | item value                  | Number(25)                   |               | item id                                                    |
|             | inventory quantity          | Number(12)                   |               | total units or total weight                                |
|             | location description        | Char(150)                    |               | NULL                                                       |
| FTAIL       | file type record descriptor | Char(5)                      | FTAIL         | hardcode 'FTAIL'                                           |
|             | file line identifier        | Number(10)                   |               | Id of current line being processed, internally incremented |
|             | file record count           | Number(10)                   |               | Number of detail records.                                  |



## stkdlly (Stock Count Shrinkage Update)

### Functional Area

Stock Ledger

### Module Affected

STKDLY.PC

### Design Overview

This program processes the 'Unit & Dollar' type of stock count that the user has submitted for processing for the stock ledger. The main functions are to calculate actual shrinkage amount that is used to correct the book stock value on the stock ledger and to calculate a shrinkage rate. A system option indicator (CLOSE\_MTH\_WITH\_OPN\_CNT\_IND) is used to determine whether or not the current fiscal month is allowed to be closed while containing an open Unit and Dollar stock count.

If the indicator is No (that is, fiscal month may not be closed with existing open Unit and Dollar stock counts), the program raises a fatal error if open stock counts are found within the current fiscal month. If no open stock counts are found within the current fiscal month, the program calculates the book stock value for the current months scheduled stock counts. It then compares the book stock value to the actual stock value as reported on the stock count. These values, and their difference are used to update month data records. Values such as shrinkage, book stock, and actual stock are modified as a consequence. Week data are similarly updated; since it is always the current month being processed, current half-year data records for inter-stock-take and sales can be updated with these values as well.

For more information, see the chapter "Stock Ledger Batch" in this volume of the RMS Operations Guide.

### Scheduling Constraints

| Schedule Information      | Description                         |
|---------------------------|-------------------------------------|
| Processing Cycle          | Phase 3 (daily)                     |
| Scheduling Considerations | Run before SALWEEK.PC and SALMTH.PC |
| Pre-Processing            | N/A                                 |
| Post-Processing           | N/A                                 |
| Threading Scheme          | Threaded by department              |

### Restart/Recovery

This batch program is multithreaded using the v\_restart\_dept view. The logical unit of work for this program is dept/class/location.

### Locking Strategy

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| PERIOD           | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS   | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES | Yes    | No     | No     | No     |
| STAKE_PROD_LOC   | Yes    | No     | Yes    | No     |
| STAKE_HEAD       | Yes    | No     | No     | No     |
| DEPS             | Yes    | No     | No     | No     |
| HALF_DATA_BUDGET | Yes    | No     | No     | No     |
| DAILY_DATA       | Yes    | No     | No     | No     |
| WEEK_DATA        | No     | No     | Yes    | No     |
| MONTH_DATA       | Yes    | No     | Yes    | No     |
| HALF_DATA        | No     | No     | Yes    | No     |
| DAILY_DATA_TEMP  | No     | Yes    | No     | No     |

**I/O Specification**

N/A

**stkprg (Purge Stock Count)****Functional Area**

Stock Count

**Module Affected**

STKPRG.PC

**Design Overview**

This batch program deletes records from the stock count tables with a STAKE\_HEAD.STOCKTAKE\_DATE that is less than the SYSTEM\_VARIABLES.LAST\_EOM\_START\_MONTH. The program deletes outdated records from STAKE\_HEAD and their corresponding child records on the STAKE\_SKU\_LOC and STAKE\_PROD\_LOC tables.

**Scheduling Constraints**

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | AD HOC      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| SYSTEM_VARIABLES | Yes    | No     | No     | No     |
| STAKE_LOCATION   | Yes    | No     | No     | Yes    |
| STAKE_QTY        | No     | No     | No     | Yes    |
| STAKE_CONT       | No     | No     | No     | Yes    |
| STAKE_SKU_LOC    | No     | No     | No     | Yes    |
| STAKE_PROD_LOC   | No     | No     | No     | Yes    |
| STAKE_PRODUCT    | No     | No     | No     | Yes    |
| STAKE_HEAD       | Yes    | No     | No     | Yes    |

**I/O Specification**

N/A

**stkschedxpld (Scheduled Stock Count Explode)****Functional Area**

Stock Count

## Module Affected

STKSCHEDXPLD.PC

## Design Overview

This batch program (STKSCHEDXPLD.PC) is used to create the scheduled stock counts for the location. It finds all the stock count schedules, which are set for the location using the SYSTEM\_OPTIONS.STAKE\_REVIEW\_DAYS. The schedule can be set to fire on daily basis or else the user can specify the days (Sunday, Monday, and so on) on which the stock count can be created. In essence, the users specify the cycles such as “every third Monday” and “every second Tuesday and Thursday.”

If the count is a Unit Only Count, then the item list is specified in the detail record. In this case, the item list is exploded out to the SKU, and every SKU is added to the count/item/location tables. The SKU which belongs to the item list in a department/class/subclass is added to the count/item/location tables.

If the count is a unit and amount count, then the department/class/subclass is fully exploded out to the subclass level. All the department/class/subclass combinations is added to the count/location/subclass tables.

If the location is a location list, then the list is exploded out to the locations before the detail record is processed.

If the location is a warehouse, then the already existing warehouse stores not be added to current count and also the processing is moved to the next location.

If the location is a store and any simple pack exists for an item list on the count, then the simple pack and the item list is added for the count. If the simple pack is on the count, then its component SKU stores also be added along with any other simple packs containing the SKU which is not already in the count.

## Scheduling Constraints

| Schedule Information      | Description                                       |
|---------------------------|---------------------------------------------------|
| Processing Cycle          | Phase 0 - Daily                                   |
| Scheduling Considerations | Run before STKXPLD.PC.                            |
| Pre-Processing            | N/A                                               |
| Post-Processing           | N/A                                               |
| Threading Scheme          | Multi-threaded by location (store and warehouse). |

## Restart/Recovery

The logical unit of work for this module is schedule, location. The changes are posted when the commit\_max\_ctr value is reached.

## Locking Strategy

N/A

## Security Considerations

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| STAKE_SCHEDULE     | Yes    | No     | Yes    | No     |
| V_RESTART_STORE_WH | Yes    | No     | No     | No     |
| PERIOD             | Yes    | No     | No     | No     |
| CODE_DETAIL        | Yes    | No     | No     | No     |
| STAKE_HEAD         | No     | Yes    | No     | No     |
| STAKE_LOCATION     | No     | Yes    | No     | No     |
| STAKE_PRODUCT      | No     | Yes    | No     | No     |
| STAKE_PROD_LOC     | No     | Yes    | No     | No     |
| STAKE_SKU_LOC      | Yes    | Yes    | No     | No     |
| ITEM_MASTER        | Yes    | No     | No     | No     |
| DEPS               | Yes    | No     | No     | No     |
| SUBCLASS           | Yes    | No     | No     | No     |
| PACKITEM           | Yes    | No     | No     | No     |
| ITEM_LOC           | Yes    | No     | No     | No     |
| SKULIST_DETAIL     | Yes    | No     | No     | No     |
| LOC_LIST_DETAIL    | Yes    | No     | No     | No     |
| LOCATION_CLOSED    | Yes    | No     | No     | No     |
| COMPANY_CLOSED     | Yes    | No     | No     | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**stkupd (Stock Count Snapshot Update)****Functional Area**

Stock Count

**Module Affected**

STKUPD.PC

## Design Overview

This batch program retrieves the stock-on-hand for each item/location record on a scheduled stock count and uses it to update the snapshot on-hand quantity on the corresponding STAKE\_SKU\_LOC record.

## Scheduling Constraints

| Schedule Information      | Description                                  |
|---------------------------|----------------------------------------------|
| Processing Cycle          | Phase 3(daily)                               |
| Scheduling Considerations | stkxpdl.pc should run prior to this program. |
| Pre-Processing            | prepost stkupd pre                           |
| Post-Processing           | prepost stkupd post                          |
| Threading Scheme          | Threaded by location                         |

## Restart/Recovery

This program is multithread using the v\_restart\_all\_locations view. The logical unit of work is an item/location.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |
| PERIOD         | Yes    | No     | No     | No     |
| STAKE_SKU_LOC  | Yes    | No     | Yes    | No     |
| STAKE_HEAD     | Yes    | No     | No     | No     |
| ITEM_LOC_SOH   | Yes    | No     | No     | No     |

## I/O Specification

N/A

# stkupld (Upload Stock Count)

## Functional Area

Inventory Management – Stock count

## Module Affected

STKUPLD.PC

## Design Overview

This module uploads actual count data from the selected store or physical warehouse to STAKE\_SKU\_LOC. The module is designed to upload a flat file layout that contains stock count data prepared by the retailer. For a physical warehouse in a multi-channel environment, STKUPLD.PC calls RMS' distribution library to apportion quantities to the virtual warehouses in RMS.

## Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                           |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 1 (Daily)                                                                                                                                                       |
| Scheduling Considerations | Run after LIKSTKUP.PC. The uploaded file sent by the warehouse management system is first translated by LIFSTKUP.PC before STKUPLD.PC inputs the file for processing. |
| Pre-Processing            | N/A                                                                                                                                                                   |
| Post-Processing           | N/A                                                                                                                                                                   |
| Threading Scheme          | N/A                                                                                                                                                                   |

## Restart/Recovery

The logical unit of work for the stock take upload module is a count of discrete inventory transactions. Each record is uniquely identified by a location and item. The logical unit of work is defined as a number of these transaction records, determined by the commit\_max\_ctr field on the RESTART\_CONTROL table.

The file records are grouped in numbers equal to the commit\_max\_ctr. After all records in a given read are processed (or rejected), the restart commit logic and restart file writing logic is called, after which the following group of file records are read and processed. The commit logic stores save the current file pointer position in the input file and any application image information (for example, record and reject counters) and commit all database transactions. The file writing logic stores append the temporary holding files to the final output files.

The commit\_max\_ctr field should be set to prevent excessive rollback space usage and to reduce the overhead of file I/O. The recommended commit counter setting is 10,000 records (subject to change based on experimentation).

Error handling stores recognize three levels of record processing: process success, non-fatal errors, and fatal errors. Item level validation stores occur on all fields before table processes are initiated. If all field-level validations return successfully, inserts and updates are allowed. If a non-fatal error is produced, the remaining fields are validated but the record is rejected and written to the reject file. If a fatal error is returned, file processing stores end immediately. A restart is initiated from the file pointer position saved in the restart\_bookmark string at the time of the last commit point that was reached during file processing.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| STK_FILE_STG       | Yes    | Yes    | No     | No     |
| STAKE_SKU_LOC      | Yes    | Yes    | Yes    | No     |
| STK_SSL_TEMP       | Yes    | Yes    | No     | No     |
| STAKE_QTY          | Yes    | Yes    | Yes    | No     |
| WH                 | Yes    | No     | No     | No     |
| ITEM_LOC_SOH       | Yes    | No     | No     | No     |
| ITEM_LOC           | Yes    | No     | No     | No     |
| STK_SSL_TEMP       | Yes    | Yes    | No     | No     |
| STK_XFORM_TEMP     | Yes    | Yes    | No     | No     |
| STAKE_PROD_LOC     | Yes    | No     | No     | No     |
| STAKE_PRODUCT      | Yes    | No     | No     | No     |
| ITEM_MASTER        | Yes    | No     | No     | No     |
| STAKE_PROD_LOC     | Yes    | No     | No     | No     |
| ITEM_XFORM_DETAIL  | Yes    | No     | No     | No     |
| ITEM_XFORM_HEAD    | Yes    | No     | No     | No     |
| STK_XFORM_ORD_TEMP | Yes    | Yes    | No     | No     |
| STAKE_LOCATION     | Yes    | Yes    | No     | No     |
| PARTNER            | Yes    | No     | No     | No     |
| STAKE_HEAD         | Yes    | No     | No     | No     |

**I/O Specification****Input File Layout**

| Record Name | Field Name           | Field Type | Default Value | Description              |
|-------------|----------------------|------------|---------------|--------------------------|
| File Header | File head descriptor | Char(5)    | FHEAD         | Describes file line type |



| Record Name        | Field Name             | Field Type | Default Value | Description                                                                  |
|--------------------|------------------------|------------|---------------|------------------------------------------------------------------------------|
|                    | file line identifier   | Number(10) | 0000000001    | Id of current line being processed                                           |
|                    | File Type              | Char(4)    | STKU          | Identifies the file type                                                     |
|                    | File create date       | Char(14)   |               | VDATE in YYYYMMDDHH24MISS format                                             |
|                    | Stock take date        | Char(14)   |               | Date on which stock count stores take place in YYYYMMDDHHMISS format         |
|                    | Cycle count            | Number (8) |               | Unique number to identify the stock count                                    |
|                    | Location Type          | Char(1)    |               | Location type<br>Accepted Value<br>'S','W','E'                               |
|                    | Location               | Number(10) |               | The location where stock count stores occur                                  |
| Transaction Record | File record descriptor | Char(5)    | FDETL         | Describes file line type                                                     |
|                    | Line Number            | Number(10) |               | Sequential file line number                                                  |
|                    | Item type              | Char(3)    | ITM           | Item type ,Hardcode value                                                    |
|                    | Item value             | Char(25)   |               | Unique identifier for item                                                   |
|                    | Inventory quantity     | Number(12) |               | Total quantity count*10000 (4 implied decimal places) for store or warehouse |
|                    | Location description   | Char(150)  |               | Description of inventory location                                            |
| FTAIL              | File record descriptor | Char(5)    | FTAIL         | Marks end of file                                                            |
|                    | File line identifier   | Number(10) |               | Id of current line being processed, internally incremented                   |
|                    | File record count      | Number(10) | .             | Number of detail records                                                     |

## stkvar (Stock Count Stock on Hand Updates)

### Functional Area

Stock Count

### Module Affected

STKVAR.PC

### Design Overview

This batch module updates the stock on hand in STAKE\_PROD\_LOC, along with ITEM\_LOC\_SOH table. It also computes the total cost and total retail .It checks the system VAT indicator, the indicator for stock ledger VAT, the class level VAT indicator, and the indicator for retail inclusion of VAT indicator for the class to determine if VAT needs to be added on, stripped off, or neither before updating the STAKE\_PROD\_LOC table.

### Scheduling Constraints

| Schedule Information      | Description          |
|---------------------------|----------------------|
| Processing Cycle          | Phase 1(Daily)       |
| Scheduling Considerations | N/A                  |
| Pre-Processing            | N/A                  |
| Post-Processing           | N/A                  |
| Threading Scheme          | Threaded by location |

### Restart/Recovery

The logical unit of work for this program is item, loc\_type and location. This program is multithread using the v\_restart\_all\_locations view. After the commit\_max\_ctr number of rows is processed, intermittent commits are done to the database and the item/location information is written to restart tables for restart/recovery.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |
| PERIOD         | Yes    | No     | No     | No     |

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| ITEM_XFORM_HEAD   | Yes    | No     | No     | No     |
| ITEM_XFORM_DETAIL | Yes    | No     | No     | No     |
| STAKE_SKU_LOC     | Yes    | No     | Yes    | No     |
| STAKE_CONT        | Yes    | No     | No     | Yes    |
| STAKE_HEAD        | Yes    | No     | No     | No     |
| STAKE_CONT_TEMP   | Yes    | Yes    | No     | Yes    |
| STAKE_PROD_LOC    | Yes    | No     | Yes    | No     |
| WH                | Yes    | No     | No     | No     |
| CLASS             | Yes    | No     | No     | No     |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| ITEM_LOC_SOH      | Yes    | No     | Yes    | No     |
| ITEM_SUPP_COUNTRY | Yes    | No     | No     | No     |
| EDI_DAILY_SALES   | No     | No     | Yes    | No     |
| TRAN_DATA         | No     | Yes    | No     | No     |
| NWP               | No     | Yes    | Yes    | No     |
| NWP_FREEZE_DATE   | Yes    | No     | No     | No     |
| STAKE_QTY         | Yes    | No     | No     | No     |
| STAKE_LOCATION    | Yes    | No     | No     | No     |
| STAKE_PRODUCT     | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**stkxpld (Stock Count Explode)****Functional Area**

Stock Count

**Module Affected**

STKXPLD.PC

**Design Overview**

This batch program, in anticipation of the stock count, populates the stocktake tables with department-class-subclass relationship for all items to be counted for the chosen location.

## Scheduling Constraints

| Schedule Information      | Description                                                        |
|---------------------------|--------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (Daily)                                                    |
| Scheduling Considerations | This batch should run prior to the prepost pre job for stkup1d.pc. |
| Pre-Processing            | N/A                                                                |
| Post-Processing           | N/A                                                                |
| Threading Scheme          | Threaded by location                                               |

## Restart/Recovery

This batch program is multithreaded using the v\_restart\_all\_locations view. The logical unit of work for this program is a cycle count/location.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| PERIOD            | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS    | Yes    | No     | No     | No     |
| STAKE_LOCATION    | Yes    | No     | No     | No     |
| STAKE_HEAD        | Yes    | No     | No     | No     |
| STAKE_SKU_LOC     | Yes    | Yes    | No     | No     |
| STAKE_PROD_LOC    | Yes    | Yes    | No     | No     |
| STAKE_PRODUCT     | Yes    | No     | No     | No     |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| ITEM_LOC          | Yes    | No     | No     | No     |
| ITEM_LOC_SOH      | Yes    | No     | No     | No     |
| ITEM_XFORM_HEAD   | Yes    | No     | No     | No     |
| ITEM_XFORM_DETAIL | Yes    | No     | No     | No     |
| SUBCLASS          | Yes    | No     | No     | No     |

## I/O Specification

N/A

---

## Stock Ledger Batch

### Overview

The stock ledger holds financial data that allows you to monitor your company's performance. It incorporates financial transactions related to merchandising activities, including sales, purchases, transfers, and markdowns; and is calculated weekly or monthly. The stock ledger accounts for inventory in buckets (how much inventory was returned, how much damaged, and so on). This overview describes how the stock ledger is set up, the accounting methods that impact stock ledger calculations, the primary stock ledger tables, and the batch programs and PL/SQL packages that process data held on the tables.

---

**Note:** For additional information about stock ledger transaction posting, see the chapter "Sales Posting Batch" in this volume of the RMS Operations Guide.

---

### Stock Ledger Set Up and Accounting Methods

The operation of the stock ledger is dependent upon a number of options that you choose for your implementation of RMS. To understand how your company uses the stock ledger, you can examine the settings that are described here.

The stock ledger is implemented at the subclass level and supports both the retail and cost methods of accounting. The method of accounting may vary by department and is set on the department (DEPS) table in the profit\_calc\_type column. The '1' setting indicates that profit is calculated by direct cost. The '2' setting indicates that profit is calculated by retail inventory.

If you select the cost method of accounting, two options are available: average cost or standard cost. The chosen option is represented on the SYSTEM\_OPTIONS table in the std\_av\_ind column, where the standard cost option is indicated by the 'S' setting, and the average cost option is indicated by the 'A' setting. The selected option then applies to all departments that use the cost method stock ledger option.

If you select the retail method of accounting, you can choose to implement the retail components of all transactions either to include value-added tax (VAT) or to exclude VAT. You accomplish through a system-level option vat\_ind on the SYSTEM\_OPTIONS table.

---

**Note:** If the value-added tax (VAT) system option is enabled in RMS, rolled-up stock ledger data values for the retail accounting method include value-added tax.

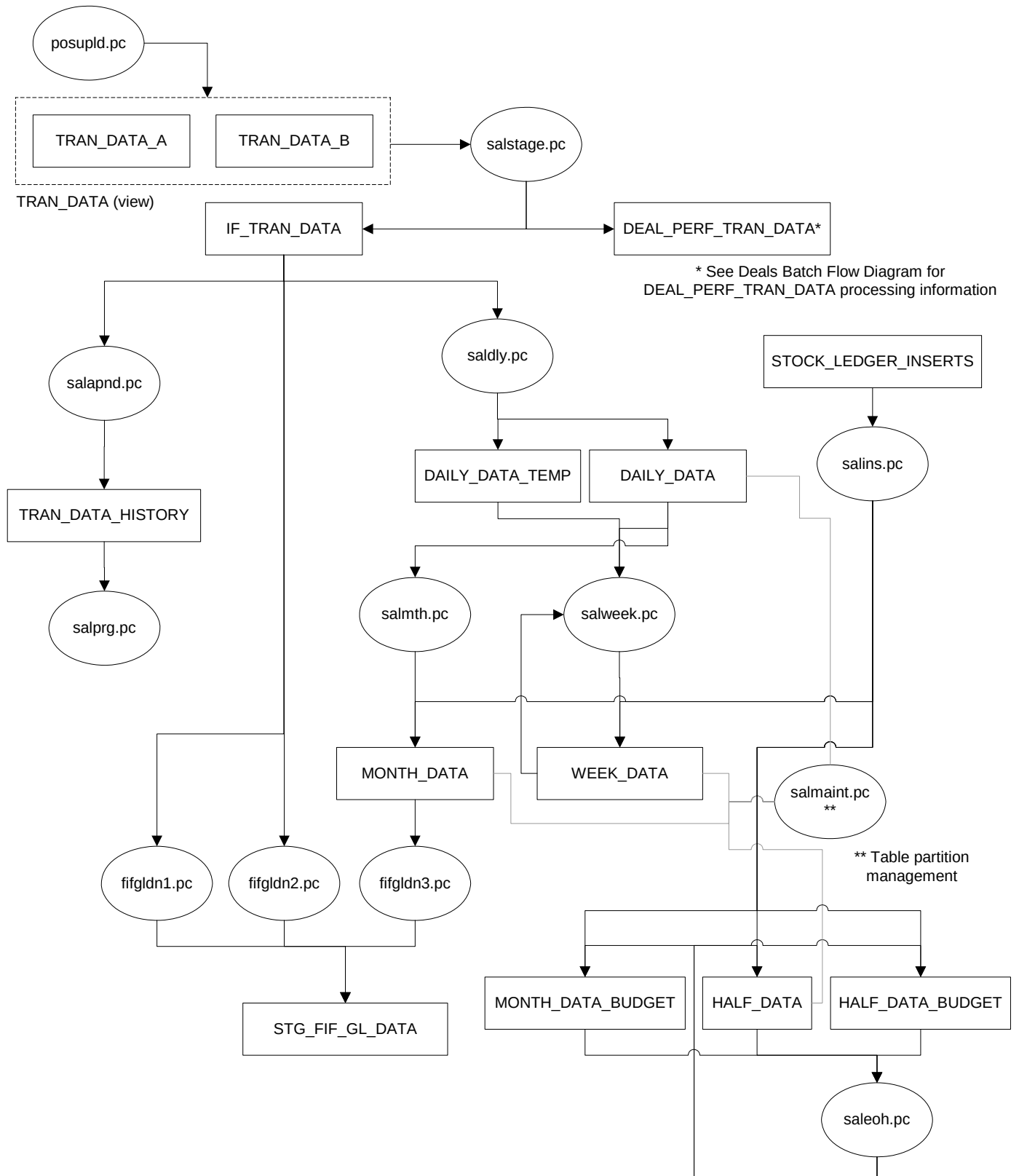
---

For sales history purposes, history is maintained based on the calendar that you choose. If your company uses the 4-5-4 calendar, sales history is tracked weekly. If you use the Gregorian (or 'normal') calendar, sales history is tracked monthly. The calendar setting is held on the SYSTEM\_OPTIONS table in the calendar\_454\_ind column.

---

**Note:** The following diagram does not include the impact of deals. For more information, see the chapter “Deals Maintenance Batch” in this volume of the RMS Operations Guide.

---



Stock Ledger Data Flow And Batch Modules



## Stock Counts and Budget Shrinkage

If a stock count has occurred during a week or month, the stock count batch module STKDLY.PC stores update WEEK\_DATA and MONTH\_DATA with the actual shrinkage calculated from stock count results. This actual shrinkage is used to adjust the inventory when SALWEEK.PC and SALMTH.PC run. In order to calculate shrinkage, the budget shrinkage indicator bud\_shrink\_ind must be enabled on the SYSTEMS\_OPTION table.

Budgeted shrinkage is calculated using the budgeted shrinkage percent (stored on the HALF\_DATA\_BUDGET table) multiplied by sales at retail or at cost, depending on whether retail or cost accounting method is used, respectively.

## PL/SQL Packages

A number of stock ledger batch modules call functions contained in a PL/SQL package named STKLEDGR\_ACCTING\_SQL to perform financial calculations. Functions in this package include:

- COST\_METHOD\_CALC – performs calculations for cost method of accounting such as closing stock, book stock, and gross margin
- RETAIL\_METHOD\_CALC – performs calculations for retail method of accounting such as closing stock, book stock, and gross margin
- COST\_METHOD\_CALC\_RETAIL – calculates the ending inventory value at retail for the cost method

## End of Year (NWP) Inventory

If necessary for legal or organizational reasons, a retailer can determine the end of year inventory values in each store as of December 31st of the previous year. To determine the correct end of year inventory value for this report, the following process occurs:

1. A stock count is performed for every store in the early part of the year.
2. The variance determined in this count is applied to the book stock as of the end of the year to determine more accurate end of year inventory values.
3. End of the year stock units are re-valued based on the last received cost for the items at each location during the previous year.

The end of year value of the inventory, after the stock count adjustments and revaluation process based on the lower of Weighted Average Cost or Last Received Cost, is the value that is used to report the end of year inventory.

To accomplish this, RMS uses a table, on which it holds a record for every active item/location combination for each year. Items on this table are non-pack items held at the transaction level.

New records are added to this table through several processes:

- Each time a receipt is recorded in RMS, the receipt process determines if there is currently a record for the item/location combination for the year of the receipt on the table, and if there is not, a new record is added to the table.
- When the end of year NWP snapshot process runs, it takes a snapshot of stock and weighted average cost (WAC) for every item/location combination currently holding stock. If there is not a record already on the NWP table for an item/location/year combination in the snapshot, a new record is added for that item/location/year combination.

- When an end of year NWP stock count is processed and variances are posted to the NWP table, if there is not an item/location/year record on the NWP table for the variance record, a new record is added to the NWP table for that item/location/year.

Records on this table are updated by several processes: the receipt process, NWP end of year snapshot process, and the NWP stock count process all update records on this table when a record for the item/location/year combination already exists on this table. The receiver cost adjustment process and receiver unit adjustment process also update records on this table if the item/PO/shipment record exists on this table.

Retailers indicate whether or not they use this NWP processing in the SYSTEM\_OPTIONS table.

## Batch Design Summary

The following batch designs are included in this functional area:

- NWPPURGE.PC (End of Year Inventory Position Purge)
- NWPYEAREND.PC (End of Year Inventory Position Snapshot)
- SALAPND.PC (Stock Ledger Append)
- SALDLY.PC (Daily Stock Ledger)
- SALEOH.PC (End of Half Stock Ledger Processing)
- SALINS.PC (Stock Ledger and Budget Tables Insert)
- SALMAINT.PC (Stock Ledger Table Maintenance)
- SALMTH.PC (Monthly Stock Ledger Pprocessing)
- SALPRG.PC (Purge Stock Ledger Ttransactions)
- SALSTAGE.PC (Stock Ledger Stage)
- SALWEEK.PC (Weekly Stock Ledger Processing)
- WASTEADJ.PC (Wastage Adjustment)

## nwppurge (End Of Year Inventory Position Purge)

### Functional Area

Stock Ledger

### Module Affected

NWPPURGE.PC

### Design Overview

This program purges the records from the table NWP after a certain amount of years have passed. The number of years is a configurable parameter setup in SYSTEM\_OPTIONS.nwp\_retention\_period.

### Scheduling Constraints

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad-Hoc      |
| Scheduling Considerations | N/A         |

| Schedule Information | Description |
|----------------------|-------------|
| Pre-Processing       | N/A         |
| Post-Processing      | N/A         |
| Threading Scheme     | N/A         |

**Restart/Recovery**

Restart/recovery is not applicable, but the records are committed based on the commit max counter setup in the restart control table.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| NWP   | Yes    | No     | No     | Yes    |

**I/O Specification**

NA

**nwpyearend (End of Year Inventory Position Snapshot)****Functional Area**

Stock count

**Module Affected**

NWPYEAREND.PC

**Design Overview**

This program takes a snapshot of the item's stock position and cost at the end of the year. When the end of year NWP snapshot process runs, it takes a snapshot of stock and weighted average cost (WAC) for every item/location combination currently holding stock. If there is not a record already on the NWP table for an item/location/year combination in the snapshot, a new record is added for that item/location/year combination.

### Scheduling Constraints

| Schedule Information      | Description                                          |
|---------------------------|------------------------------------------------------|
| Processing Cycle          | Phase 4 (Yearly)                                     |
| Scheduling Considerations | Needs to run on the last day of the year in phase 4. |
| Pre-Processing            | N/A                                                  |
| Post-Processing           | N/A                                                  |
| Threading Scheme          | Multithreaded by store_wh                            |

### Restart/Recovery

The logical unit of work for this program is set at the location/item level. Threading is done by supplier using the v\_restart\_store\_wh view to thread properly.

The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The changes are posted when the commit\_max\_ctr value is reached and the value of the counter is subject to change based on implementation.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| NWP_FREEZE_DATE | Yes    | No     | No     | No     |
| ITEM_MASTER     | Yes    | No     | No     | No     |
| NWP             | Yes    | Yes    | Yes    | No     |
| ITEM_LOC_SOH    | Yes    | No     | No     | No     |

### I/O Specification

N/A

## salapnd (Stock Ledger Append)

### Functional Area

Stock Ledger

**Module Affected**

SALAPND.PC

**Design Overview**

The purpose of this program is to move data from the transaction-staging table into the historical transaction table. This requires placing a lock on the staging table to ensure that no new data is added to it while the movement is occurring, moving the data to the historical table, and finally releasing the lock on the staging table.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                                                      |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3                                                                                                                                                                          |
| Scheduling Considerations | Should occur after all extractions have completed                                                                                                                                |
| Pre-Processing            | SALSTAGE, all extraction, and all processing                                                                                                                                     |
| Post-Processing           | N/A                                                                                                                                                                              |
| Threading Scheme          | Threading is implicit through the use of the Oracle Parallel Query Option. The insert/select query should be tuned for each specific environment to achieve the best throughput. |

**Restart/Recovery**

N/A

**Locking Strategy**

This batch program locks the table IF\_TRAN\_DATA in EXCLUSIVE mode with NOWAIT option specified and releases the lock once all the records are read and appended to TRAN\_DATA\_HISTORY.

**Security Considerations**

N/A

**Performance Considerations**

The degree of parallelism defined in the program's insert/select query should be tuned for each run-time environment to achieve the best throughput. The default value is set at 4.

**Key Tables Affected**

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| PERIOD            | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES  | Yes    | No     | No     | No     |
| IF_TRAN_DATA      | Yes    | No     | No     | No     |
| TRAN_DATA_HISTORY | No     | Yes    | No     | No     |

**I/O Specification**

N/A

**saldly (Sales Daily)****Functional Area**

Stock Ledger

**Module Affected**

SALDLY.PC

**Design Overview**

This module rolls up transaction data on IF\_TRAN\_DATA to the dept/class/subclass/location/transaction date/currency level.

The rolled-up transactions are used to update applicable records on DAILY\_DATA based on the transaction type. A new record is inserted if no record exists for the transaction.

**Scheduling Constraints**

| Schedule Information      | Description                                                  |
|---------------------------|--------------------------------------------------------------|
| Processing Cycle          | Phase 3 (daily)                                              |
| Scheduling Considerations | N/A                                                          |
| Pre-Processing            | Run SALSTAGE to move records from TRAN_DATA to IF_TRAN_DATA. |
| Post-Processing           | N/A                                                          |
| Threading Scheme          | Threaded by department                                       |

**Restart/Recovery**

The logical unit of work is department/class/subclass. This batch program is multithreaded using the v\_restart\_dept view.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| PERIOD           | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES | Yes    | No     | No     | No     |

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| IF_TRAN_DATA    | Yes    | No     | No     | No     |
| DAILY_DATA      | Yes    | Yes    | Yes    | No     |
| DAILY_DATA_TEMP | No     | Yes    | No     | No     |
| STORE           | Yes    | No     | No     | No     |
| WH              | Yes    | No     | No     | No     |
| PARTNER         | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS  | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**saleoh (End of Half Stock Ledger Processing)****Functional Area**

Stock Ledger

**Module Affected**

SALEOH.PC

**Design Overview**

This module purges rows on MONTH\_DATA\_BUDGET, and HALF\_DATA\_BUDGET that are three halves old (18 months or older) and performs the following inserts.

- Inserts one row into HALF\_DATA for each subclass/location combination for the next half
- Inserts six rows (one for every month of the half) into MONTH\_DATA\_BUDGET for each department/location for next year's half
- Inserts one row into HALF\_DATA\_BUDGET for each department/location for next year's half

This program also rolls up the inter\_stocktake\_shrink\_amt and inter\_stocktake\_sales\_amt from the HALF\_DATA table at the department/location level for this half and calculates the shrinkage\_pct to insert into HALF\_DATA\_BUDGET for the next year's half.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                                                                                                   |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3                                                                                                                                                                       |
| Scheduling Considerations | Run at the end of the half, after the monthly process is completed for month six (6) of the current half, and before the SALMTH process for the first month of the next half. |
| Pre-Processing            | N/A                                                                                                                                                                           |

| Schedule Information | Description             |
|----------------------|-------------------------|
| Post-Processing      | N/A                     |
| Threading Scheme     | Threaded by department. |

### Restart/Recovery

There is no main driving cursor for this program. The different functions of this batch program have their own driving cursors. All the driving cursors are threaded by department using the v\_restart\_dept view.

The logical unit of work (LUW) for the delete functions is a half number while the different insert functions have the following LUWs

- half\_data() – dept/class/subclass/location
- month\_data\_budget() – dept/location
- half\_data\_budget() – dept/location

Data is committed every time the number of rows processed exceeds commit\_max\_ctr.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS    | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES  | Yes    | No     | No     | No     |
| MONTH_DATA_BUDGET | Yes    | Yes    | No     | Yes    |
| HALF_DATA         | Yes    | Yes    | No     | No     |
| HALF_DATA_BUDGET  | Yes    | Yes    | No     | Yes    |

### I/O Specification

N/A

## salins (Stock Ledger and Budget Tables Insert)

### Functional Area

Stock Ledger

### Module Affected

SALINS.PC



## Design Overview

This module populates the stock ledger and budget tables: WEEK\_DATA, MONTH\_DATA, HALF\_DATA, MONTH\_DATA\_BUDGET, and HALF\_DATA\_BUDGET for each subclass-location or department-location combination in RMS whenever a new location, department, or subclass is added to the system.

## Scheduling Constraints

| Schedule Information      | Description     |
|---------------------------|-----------------|
| Processing Cycle          | Phase 0 (daily) |
| Scheduling Considerations | N/A             |
| Pre-Processing            | N/A             |
| Post-Processing           | N/A             |
| Threading Scheme          | N/A             |

## Restart/Recovery

The logical unit of work (LUW) is a location, type\_code, dept, class, and subclass combination. Changes are committed to the database once the commit\_max\_counter is reached for the LUW. Note that the number of database changes varies depending on the type\_code,

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS       | Yes    | No     | No     | No     |
| STOCK_LEDGER_INSERTS | Yes    | No     | No     | Yes    |
| STORE                | Yes    | No     | No     | No     |
| WH                   | Yes    | No     | No     | No     |
| PARTNER              | Yes    | No     | No     | No     |
| WEEK_DATA            | No     | Yes    | No     | No     |
| MONTH_DATA           | No     | Yes    | No     | No     |
| HALF_DATA            | No     | Yes    | No     | No     |
| MONTH_DATA_BUDGET    | No     | Yes    | No     | No     |

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| HALF_DATA_BUDGET | No     | Yes    | No     | No     |

### **I/O Specification**

N/A

## **salmaint (Stock Ledger Table Maintenance)**

### **Functional Area**

Stock Ledger

### **Module Affected**

SALMAINT.PC

### **Design Overview**

This module is run as either salmaint pre or salmaint post. The salmaint pre functionality adds partitions to the HALF\_DATA, DAILY\_DATA, WEEK\_DATA and MONTH\_DATA tables. The salmaint post functionality drops partitions or purges the above tables (if the table is not partitioned) for an old half.

### **Scheduling Constraints**

| Schedule Information      | Description |
|---------------------------|-------------|
| Processing Cycle          | Ad-Hoc      |
| Scheduling Considerations | N/A         |
| Pre-Processing            | N/A         |
| Post-Processing           | N/A         |
| Threading Scheme          | N/A         |

### **Restart/Recovery**

N/A

### **Locking Strategy**

N/A

### **Security Considerations**

N/A

### **Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS   | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES | Yes    | No     | No     | No     |
| HALF_DATA        | No     | No     | No     | Yes    |
| DAILY_DATA       | No     | No     | No     | Yes    |
| WEEK_DATA        | No     | No     | No     | Yes    |
| MONTH_DATA       | No     | No     | No     | Yes    |

**I/O Specification**

N/A

**salmth (Monthly Stock Ledger Processing)****Functional Area**

Stock ledger

**Module Affected**

SALMTH.PC

**Design Overview**

The purpose of this program is to sum up the monthly transaction totals from DAILY\_DATA and calculate the closing stock and gross margin for the current month on MONTH\_DATA. The procedure varies depending on the following factors:

- Whether the retail or cost method of accounting is used depending on the setting of DEPS.profit\_calc\_type.
- Whether a stock count of Unit and Dollar type has occurred during the month, which is determined by the presence or absence of a STAKE\_PROD\_LOC row.

In addition, this program calculates shrinkage amounts.

**Scheduling Constraints**

| Schedule Information      | Description                                                                 |
|---------------------------|-----------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 (monthly)                                                           |
| Scheduling Considerations | Can run any time after end-of-month date. SALWEEK must run prior to SALMTH. |
| Pre-Processing            | N/A                                                                         |
| Post-Processing           | Prepost salmth_post                                                         |
| Threading Scheme          | Threaded by department                                                      |

**Restart/Recovery**

The logical unit of work (LUW) for this batch program is a dept/class/subclass/loc\_type/location/currency\_ind record. This batch program is threaded by department using the v\_restart\_dept view. Processed records are committed to the database after the LUW count has reached the commit\_max\_ctr.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| PERIOD           | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS   | Yes    | No     | No     | No     |
| SYSTEM_VARIABLES | Yes    | No     | No     | No     |
| STAKE_HEAD       | Yes    | No     | No     | No     |
| STAKE_PROD_LOC   | Yes    | No     | No     | No     |
| PARTNER          | Yes    | No     | No     | No     |
| STORE            | Yes    | No     | No     | No     |
| WH               | Yes    | No     | No     | No     |
| MONTH_DATA       | Yes    | Yes    | Yes    | No     |
| DAILY_DATA       | Yes    | No     | No     | No     |
| DEPS             | Yes    | No     | No     | No     |
| WEEK_DATA        | Yes    | No     | No     | No     |
| HALF_DATA_BUDGET | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**salprg (Purge Stock Ledger Transactions)****Functional Area**

Stock ledger

**Module Affected**

SALPRG.PC

## Design Overview

This program deletes the TRAN\_DATA\_HISTORY records, which are older than the user defined number of retention days, which is stored in the SYSTEM\_OPTIONS.tran\_data\_retained\_days\_no.

## Scheduling Constraints

| Schedule Information      | Description    |
|---------------------------|----------------|
| Processing Cycle          | AD-HOC (daily) |
| Scheduling Considerations | N/A            |
| Pre-Processing            | N/A            |
| Post-Processing           | N/A            |
| Threading Scheme          | N/A            |

## Restart/Recovery

N/A

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| PERIOD            | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS    | Yes    | No     | No     | No     |
| TRAN_DATA_HISTORY | No     | No     | No     | Yes    |

## I/O Specification

N/A

## salstage (Stock Ledger Stage)

### Functional Area

Stock Ledger

### Module Affected

SALSTAGE.PC

## Design Overview

This module is used in order to make the rollup and extraction of the stock ledger transaction data the most flexible. This program stores move the data on TRAN\_DATA to the IF\_TRAN\_DATA and DEAL\_PERF\_TRAN\_DATA staging tables. This stores enable the processes that are writing records to the transaction table to continue in a seamless manner, whereas the processes that roll the data up to a different level or extract the data to external systems can work without affecting batch timetables. Locking the transaction table and moving all of the data to the staging tables stores achieve this process. The original transaction table is emptied and the lock on the table is released. Before this processing occurs, the staging tables stores first be emptied to ensure that data is not processed twice.

For more information on TRAN\_DATA codes, see the chapter, “General Ledger Batch” in this volume of the RMS Operations Guide.

## Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                  |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3, daily                                                                                                                                                               |
| Scheduling Considerations | This module should run after POSUPLD.PC and RPMMOVAVG.PC, but before SALDLY.PC, SALWEEK.PC and SALAPND.PC. Within the deal cycle, it should run before DEALACT.PC.           |
| Pre-Processing            | N/A                                                                                                                                                                          |
| Post-Processing           | N/A                                                                                                                                                                          |
| Threading Scheme          | Threading is implicit via the use of the Oracle Parallel Query Option. The insert/select query should be tuned for each specific environment to achieve the best throughput. |

## Restart/Recovery

N/A

## Locking Strategy

TRAN\_DATA\_A and TRAN\_DATA\_B tables are locked in order to move all the data from these tables to a staging table.

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table        | Select | Insert | Update | Delete |
|--------------|--------|--------|--------|--------|
| IF_TRAN_DATA | No     | Yes    | No     | Yes    |
| TRAN_DATA_A  | Yes    | Yes    | No     | Yes    |

| Table               | Select | Insert | Update | Delete |
|---------------------|--------|--------|--------|--------|
| TRAN_DATA_B         | Yes    | Yes    | No     | Yes    |
| DEAL_PERF_TRAN_DATA | No     | Yes    | No     | Yes    |

**I/O Specification**

N/A

**salweek (Sales Weekly)****Functional Area**

Stock Ledger

**Module Affected**

SALWEEK.PC

**Design Overview**

This program processes all weeks that are in the month for which month-end process is not run, up to the current week. It rolls up data on DAILY\_DATA, DAILY\_DATA\_TEMP and WEEK\_DATA\_TEMP to the corresponding dept/class/subclass/location/half-month/week/currency level and inserts into WEEK\_DATA.

The closing stock value and gross margin are calculated by calling the appropriate package function, based on the accounting method (for example, profit calculation type) chosen for the department. Closing stock value for a processed week becomes opening stock value for the next week. A WEEK\_DATA row for next week is inserted if it does not already exist.

If a stock count occurs during a certain week, STKDLY.PC would have already updated that week's data and retail fields with the difference between book stock value and actual results of stock count. In addition, this program calculates a shrinkage amount and then used to reduce the closing stock for each week processed.

This process should run at the end of each week after SALDLY.PC, with no requirements that all the transaction data be collected for that week, because SALWEEK.PC accepts 'late' transactions from previous weeks. SALWEEK.PC updates previous weeks as long as the transaction date belongs to a month that is not 'closed'. In other words, it updates all weeks since the LAST\_EOM\_DATE.

After SALWEEK.PC runs, PREPOST.PC runs its SALWEEK\_POST function to update the appropriate system variables. SALWEEK.PC also runs immediately prior to running the monthly process (SALMTH.PC) to ensure that WEEK\_DATA is in sync with MONTH\_DATA. No PREPOST function needs to run in this case.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                      |
|---------------------------|--------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3, weekly or whenever SALMNTH.PC is run                                                    |
| Scheduling Considerations | This program should run after SALDLY.PC, STKDLY.PC, SALAPND.PC and immediately before SALMTH.PC. |

| Schedule Information | Description                                                                                                                                                 |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Pre-Processing       | Prepost salweek pre – refresh data in salweek_c_week based on daily and weekly data. Refresh data in salweek_restart_dept table to balance thread activity. |
| Post-Processing      | Prepost salweek post – update system_variables.last_eow_date if running on an eow date. Truncate DAILY_DATA_TEMP table.                                     |
| Threading Scheme     | Multithreaded on Dept                                                                                                                                       |

### Restart/Recovery

The logical unit of work is dept/class/subclass combination. A commit stores take place when number of dept/class/subclass combination records processed is equal to commit max counter in restart control table.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

Due to the transaction volumes that this program must process, array processing is heavily utilized in making Oracle system calls.

### Key Tables Affected

| Table                | Select | Insert | Update | Delete |
|----------------------|--------|--------|--------|--------|
| SALWEEK_RESTART_DEPT | Yes    | No     | No     | No     |
| SALWEEK_C_WEEK       | Yes    | No     | No     | No     |
| SALWEEK_C_DAILY      | Yes    | No     | No     | No     |
| WEEK_DATA            | Yes    | Yes    | Yes    | No     |
| PARTNER              | Yes    | No     | No     | No     |
| STORE                | Yes    | No     | No     | No     |
| WH                   | Yes    | No     | No     | No     |
| DEPS                 | Yes    | No     | No     | No     |
| HALF_DATA_BUDGET     | Yes    | No     | No     | No     |

### I/O Specification

N/A



## wasteadj (Wastage adjustment)

### Functional Area

Stock ledger.

### Module Affected

WASTEADJ.PC

### Design Overview

This program stores reduce inventory of spoilage type wastage items to account for natural wastage that occurs over the shelf life of the product. This program stores affect only items with spoilage type wastage identified on ITEM\_MASTER with a waste\_type of 'SP' (spoilage). Sales type wastage is accounted for at the time of sale.

### Scheduling Constraints

| Schedule Information      | Description                                                                                                                                                                                  |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 3                                                                                                                                                                                      |
| Scheduling Considerations | This program should be scheduled to run prior to the stock count and stock ledger batch to ensure that the stock adjustment taken during the current day is credited to the appropriate day. |
| Pre-Processing            | N/A<br>.                                                                                                                                                                                     |
| Post-Processing           | N/A                                                                                                                                                                                          |
| Threading Scheme          | Threaded by store                                                                                                                                                                            |

### Restart/Recovery

The logical unit of work is an item/location. This batch program commits when the number of records processed has reached commit\_max\_ctr. If the program aborts, it restarts from the last successfully processed item /location.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table       | Select | Insert | Update | Delete |
|-------------|--------|--------|--------|--------|
| ITEM_MASTER | Yes    | No     | No     | No     |

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| ITEM_LOC       | Yes    | No     | No     | No     |
| ITEM_LOC_SOH   | Yes    | No     | Yes    | No     |
| CLASS          | Yes    | No     | No     | No     |
| INV_ADJ_REASON | Yes    | No     | No     | No     |
| INV_ADJ        | No     | Yes    | No     | No     |
| TRAN_DATA      | No     | Yes    | No     | No     |
| PERIOD         | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |

**I/O Specification**

N/A

---

## Stock Order Receipt Reconciliation Batch

### Overview

RMS receives against purchase orders and stock orders (transfers and allocations). The primary inputs to receiving processes are messages on the RIB to which RMS subscribes. Currently, RMS performs line item receiving. Carton-level receiving occurs at the distribution center or store system level. At that point the cartons are broken down and subsequently interfaced to RMS at the distribution number/carton/line item level.

When exceptions arise in the new carton level receiving processing, RMS must resolve them. One such exception is dummy carton receiving. If, for some reason, a carton number is unknown or undetectable, it may be that the part of the carton containing the label is damaged. In such a case, RMS matches up the contents of the carton with as yet unreceived cartons with the same destination, contents, and quantity. This functionality is optional. The system option `dummy_carton_ind` is available to turn this functionality on or off.

### Batch Design Summary

The following batch designs are included in this functional area:

- DUMMYCTN.PC (Dummy Carton)
- TAMPERCTN (Tampered Carton)

### dummyctn (Dummy Carton)

#### Functional Area

Receiving

#### Module Affected

DUMMYCTN.PC

#### Design Overview

this batch process scans stock orders to find not received cartons that match the dummy carton receipt (both item and quantity). If a match is found, then the dummy carton is received against the matching carton. If a match is not found, an error is written to a specific batch error file, and the record remains on the staging table. Whenever a stock order is closed, any corresponding records on the staging table are deleted. Until that point, these records may be viewed via a custom report or other mechanism.

The dummy carton receiving process stores only happen if the `dummy_carton_ind` on `system_options` is 'Y' (Yes).

#### Scheduling Constraints

| Schedule Information      | Description    |
|---------------------------|----------------|
| Processing Cycle          | Ad hoc (Daily) |
| Scheduling Considerations | N/A            |

| Schedule Information | Description |
|----------------------|-------------|
| Pre-Processing       | N/A         |
| Post-Processing      | N/A         |
| Threading Scheme     | N/A         |

### Restart/Recovery

This program deletes from the DUMMY\_CARTON\_STAGE table. The program stores restart by processing the records that remain on the DUMMY\_CARTON\_STAGE table.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected .

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| SHIPSKU_TEMP       | Yes    | Yes    | No     | Yes    |
| SHIPMENT           | Yes    | No     | Yes    | No     |
| SHIPSKU            | Yes    | No     | Yes    | No     |
| PACKITEM           | Yes    | No     | No     | No     |
| DUMMY_CARTON_STAGE | Yes    | Yes    | Yes    | Yes    |
| TSFHEAD            | Yes    | No     | No     | No     |
| ALLOC_HEADER       | Yes    | No     | No     | No     |
| TSFHEAD            | Yes    | No     | No     | No     |
| IF_ERRORS          | No     | Yes    | No     | No     |
| ALLOC_DETAIL       | No     | No     | Yes    | No     |
| SHIPITEM_INV_FLOW  | No     | No     | Yes    | No     |
| APPT_DETAIL        | No     | No     | Yes    | No     |
| DOC_CLOSE_QUEUE    | No     | Yes    | No     | No     |
| TRAN_DATA          | No     | Yes    | No     | No     |
| ITEM_LOC_SOH       | No     | Yes    | Yes    | No     |
| EDI_DAILY_SALES    | No     | No     | Yes    | No     |
| STAKE_SKU_LOC      | No     | Yes    | Yes    | No     |

| Table                          | Select | Insert | Update | Delete |
|--------------------------------|--------|--------|--------|--------|
| STAKE_PROD_LOC                 | No     | No     | Yes    | No     |
| MRT_ITEM_LOC                   | No     | No     | Yes    | No     |
| TSFDETAIL                      | No     | Yes    | Yes    | No     |
| NWP                            | No     | Yes    | Yes    | No     |
| INV_ADJ                        | No     | Yes    | No     | No     |
| SHIPITEM_INV_FLOW              | No     | Yes    | Yes    | No     |
| TSFDETAIL_CHRG                 | No     | Yes    | No     | No     |
| ITEM_LOC                       | No     | Yes    | No     | No     |
| POS_MODS                       | No     | Yes    | No     | No     |
| PRICE_HIST                     | No     | Yes    | No     | No     |
| ITEM_SUPP_COUNTRY_LOC          | No     | Yes    | Yes    | No     |
| ITEM_SUPP_COUNTRY_BRACKET_COST | No     | Yes    | Yes    | No     |
| INV_STATUS_QTY                 | No     | Yes    | Yes    | Yes    |

**I/O Specification**

N/A

**tamperctn (Tampered carton)****Functional Area**

Store Receiving

**Module Affected**

TAMPERCTN.PC

**Design Overview**

The Tampered Carton module (tamperctn.pc) is a batch program that stores match the tampered carton information in the staging table to existing shipment records. If the shipment records contain a prepack, then the batch program stores use the prepack components to compare with the items on the staging table.

**Scheduling Constraints**

| Schedule Information      | Description                                                                                      |
|---------------------------|--------------------------------------------------------------------------------------------------|
| Processing Cycle          | AD-HOC                                                                                           |
| Scheduling Considerations | This batch program should only run when the store_pack_comp_rcv_ind system option is set to 'Y'. |
| Pre-Processing            | N/A                                                                                              |

| Schedule Information | Description |
|----------------------|-------------|
| Post-Processing      | N/A         |
| Threading Scheme     | N/A         |

**Restart/Recovery**

N/A

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| SYSTEM_OPTIONS     | Yes    | No     | No     | No     |
| DUMMY_CARTON_STAGE | Yes    | No     | No     | Yes    |
| PERIOD             | Yes    | No     | No     | No     |
| ALLOC_HEADER       | Yes    | No     | No     | No     |
| SHIPMENT           | Yes    | No     | No     | No     |
| SHIPSKU            | Yes    | No     | No     | No     |
| SHIPSKU_TEMP       | Yes    | Yes    | No     | Yes    |
| PACKITEM           | Yes    | No     | No     | No     |

**I/O Specification**

N/A

---

## Store Grade Batch

### Overview

The store grade upload is designed to load store grades from an external system into RMS.

GRADUPLD.PC (store grade upload) loads data into the STORE\_GRADE\_GROUP, STORE\_GRADE and STORE\_GRADE\_STORE tables. If the store is assigned to a 'Junk' grade within the external system, then it is assigned to the 'Junk' Store in RMS. GRADUPLD.PC converts the flat file from the external system into a standard RMS batch input file.

### Batch Design Summary

The following batch design is included in this functional area:

- GRADUPLD.PC (Store Grade Upload)

### gradupld (Store Grade Upload)

#### Functional Area

Store Grade

#### Module Affected

GRADUPLD.PC

#### Design Overview

The store grade upload module is designed to load forecasting -driven store grades into RMS. Data is loaded into the STORE\_GRADE\_GROUP, STORE\_GRADE and STORE\_GRADE\_STORE tables. If the store is assigned to a 'Junk' grade within the forecasting system's grade, the store's sister store grade assignment is assigned to the 'Junk' store.

#### Scheduling Constraints

| Schedule Information      | Description                 |
|---------------------------|-----------------------------|
| Processing Cycle          | Ad Hoc                      |
| Scheduling Considerations | N/A                         |
| Pre-Processing            | N/A                         |
| Post-Processing           | N/A                         |
| Threading Scheme          | N/A – File-based processing |

## Restart/Recovery

Oracle Retail standard restart/recovery is used. The commit\_max\_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 1000 records (subject to change based on implementation).

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| Buyer             | Yes    | No     | No     | No     |
| Store             | Yes    | No     | No     | No     |
| Store_grade_group | Yes    | Yes    | No     | No     |
| Store_grade       | Yes    | Yes    | No     | No     |
| Store_grade_store | Yes    | Yes    | Yes    | No     |

## I/O Specification

### Input File Layout

| Record Name | Field Name     | Field Type | Default Value | Description              |
|-------------|----------------|------------|---------------|--------------------------|
| FHEAD       | Record type    | Char(5)    | FHEAD         | Record Identifier        |
|             | Line ID        | Number(10) | 0000000001    | Line Sequence Identifier |
|             | File name      | Char(5)    | GRADU         | File Identifier          |
| FDETL       | Record type    | Char(5)    | FDETL         | Record Identifier        |
|             | Line id        | Number(10) |               | Line Sequence Identifier |
|             | Grade Group ID | Number(8)  |               | Valid Grade Group ID     |
|             | Grade Group    | Char(120)  |               | Valid Grade Group        |
|             | Grade store    | Number(10) |               | Valid Grade store        |
|             | Grade ID       | Number(10) |               | Valid Grade ID           |
|             | Grade name     | Char(120)  |               | Valid Grade name         |
| FTAIL       | Record Type    | Char(5)    | FTAIL         | Record Identifier        |



| Record Name | Field Name | Field Type | Default Value | Description                              |
|-------------|------------|------------|---------------|------------------------------------------|
|             | Line id    | Number(10) |               | Line Sequence Identifier                 |
|             | Line Total | Number(10) |               | Total number of FDETL lines in the file. |

## Overview

The SUPMTH.PC (Supplier data amount repository) module is executed based on multiple transaction types for each department-supplier combination in the system. Its primary function is to convert daily transaction data to monthly data. After all data is converted, the daily information is deleted to reset the system for the next period by the batch module PREPOST and its supmth\_post function.

SUPMTH.PC accumulates SUP\_DATA amounts by department/supplier/transaction type and creates or updates one SUP\_MONTH row for each department/supplier combination. Based on the transaction type on SUP\_DATA, the following fields on SUP\_MONTH are updated:

- type 1 – purchases at cost (written for consignment sales and orders received at POS or online)
- type 2 – purchases at retail (written for consignment sales and orders received at POS or online)
- type 3 – claims at cost (written for claim dollars refunded on RTV orders )
- type 10 – markdowns at retail (net amount based on markdowns, markups, markdown cancellations and markup cancellations)
- type 20 – order cancellation costs (written for all supplier order cancellations)
- type 30 – sales at retail (written for consignment stock sales)
- type 40 – quantity failed (written for QC shipments with failed quantities)
- type 70 – markdowns at cost (net amount based on supplier cost markdowns)

## Batch Design Summary

The following batch design is included in this functional area:

- SUPMTH.PC (Supplier Data Amount Repository)

## supmth (Supplier Data Amount Repository)

### Functional Area

Ordering

### Module Affected

SUPMTH.PC

### Design Overview

Accumulates SUP\_DATA amounts by department/supplier/transaction type and creates or updates one SUP\_MONTH row for each department/supplier combination. Based on the transaction type on SUP\_DATA, the following fields on SUP\_MONTH are updated:

- type 1 - purchases at cost (written for consignment sales and orders received at POS or online)

- type 2 - purchases at retail (written for consignment sales and orders received at POS or online)
- type 3 - claims at cost(written for claim dollars refunded on RTV orders )
- type 10 - markdowns at retail (net amount based on markdowns, markups, markdown cancellations and markup cancellations)
- type 20 - order cancellation costs (written for all supplier order cancellations)
- type 30 - sales at retail (written for consignment stock sales)
- type 40 - quantity failed (written for QC shipments with failed quantities)

### Scheduling Constraints

| Schedule Information      | Description                                                  |
|---------------------------|--------------------------------------------------------------|
| Processing Cycle          | Phase 3 (monthly)                                            |
| Scheduling Considerations | N/A                                                          |
| Pre-Processing            | N/A                                                          |
| Post-Processing           | supmth_post(), records are truncated from the sup_data table |
| Threading Scheme          | Threaded by department                                       |

### Restart/Recovery

The logical unit of work is dept, supplier.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| SUP_DATA         | Yes    | No     | No     | No     |
| SUP_MONTH        | No     | Yes    | No     | No     |
| SYSTEM_VARIABLES | Yes    | No     | No     | No     |

### I/O Specification

N/A



## Overview

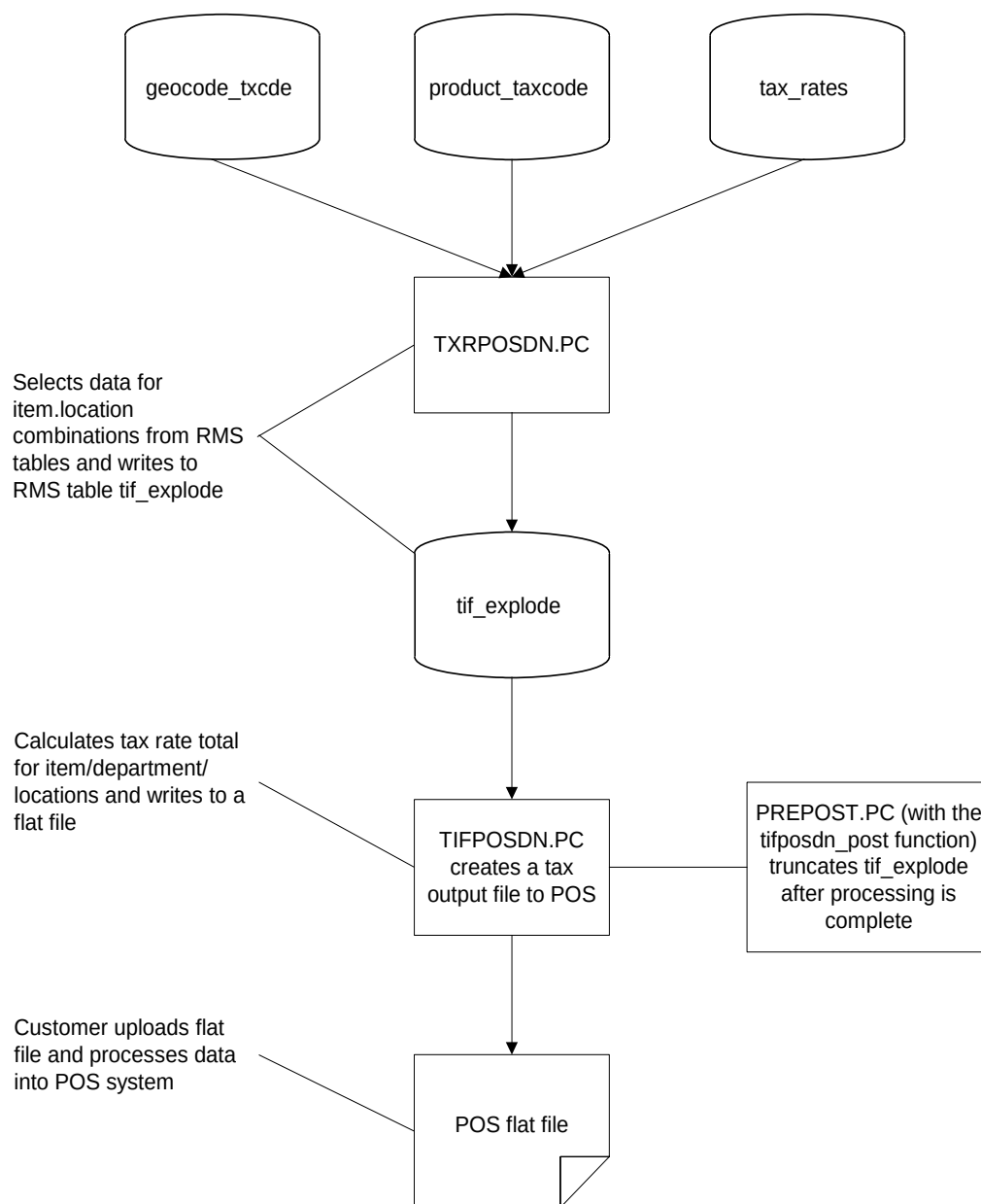
Sales tax functionality in RMS involves preparing the appropriate sales tax data for an item at a location, such as a store, and then sending that data in a file to the customer's point-of-sale system. Whenever the tax for an item at a location changes—a geocode, geocode and tax code combination, a product tax code combination, or a tax rate—RMS prepares and downloads to the location one cumulative tax rate for every affected item. The following two batch programs run daily to facilitate this process:

- TXRPOSDN.PC
- TIFPOSDN.PC

TXRPOSDN.PC processes rows off the GEOCODE\_TXCDE (GEOCODE tax code) table, PRODUCT\_TAXCODE (PRODUCT tax code) table, and the TAX\_RATES (tax rate) table. It then writes all item-location combinations to the TIF\_EXPLODE table.

Next, TIFPOSDN.PC processes data from TIF\_EXPLODE, computes a cumulative tax rate for each item-location combination, and writes the cumulative tax rate to a flat file. The flat file is then available for upload to and processing in the point-of-sale system.

The diagram below illustrates the tax rate download process.



**Tax Rate Download Process**

## Batch Design Summary

The following batch designs are included in this functional area:

- TIFPOSDN.PC (Tax Interface Point of Sale)
- TXRPOSDN.PC (Tax Rate Point of Sale Download)
- TXRTUPLD.PC (Tax Rate Upload)

## tifposdn (Tax Rate POS Download)

### Functional Area

Tax rate batch

**Module Affected**

TIFPOSDN.PC

**Design Overview**

This program processes data from TIF\_EXPLODE table, computes a cumulative tax rate for each item-location combination, and writes the cumulative tax rate to a flat file. The flat file is then available for upload to and processing in the point-of-sale system.

**Scheduling Constraints**

| Schedule Information      | Description                                |
|---------------------------|--------------------------------------------|
| Processing Cycle          | Phase 4, daily                             |
| Scheduling Considerations | N/A                                        |
| Pre-Processing            | TXRPOSDN.PC                                |
| Post-Processing           | prepost.pc (to TRUNCATE TIF_EXPLODE table) |
| Threading Scheme          | N/A                                        |

**Restart/Recovery**

The TIFPOSDN program is used to download records in the TIF\_EXPLODE table into a POS flat file. The logical unit of work for this program is set at the item/department/store combination level. Only table recovery is used. The recommended commit counter setting is 1000 records (subject to change based on implementation).

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| TIF_EXPLODE      | Yes    | No     | No     | No     |
| GEOCODE_STORE    | Yes    | No     | No     | No     |
| PRODUCT_TAX_CODE | Yes    | No     | No     | No     |
| TAX_RATES        | Yes    | No     | No     | No     |
| GEOCODE_TXCDE    | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name             | Field Type | Default Value    | Description                                           |
|-------------|------------------------|------------|------------------|-------------------------------------------------------|
| FHEAD       | File record descriptor | Char(5)    | FHEAD            | Identifies file record type.                          |
|             | Line number            | Number(10) | 0000000001       | Identifies the line number                            |
|             | Program descriptor     | Char(8)    | tifposdn         | Identifies the program.                               |
|             | Create date            | Char(14)   | File create date | YYYYMMDDHH24MISS format                               |
| THEAD       | File record descriptor | Char(5)    | THEAD            | Identifies Transaction Header                         |
|             | Line Number            | Number(10) |                  | Sequential Line Number                                |
|             | Transaction Number     | Number(10) |                  | Transaction Number                                    |
|             | Store                  | Number(10) |                  | Identifies store number                               |
| TDETL       | File Record descriptor | Char(5)    | TDETL            | Identifies transaction detail                         |
|             | Line Number            | Number(10) |                  | Sequential Line Number                                |
|             | Transaction Number     | Number(10) |                  | Transaction Number                                    |
|             | Department             | Number(4)  |                  | Department number.                                    |
|             | Item                   | Char(25)   |                  | Identifies the item.                                  |
|             | Tax Rate               | Number(8)  |                  | A new sales tax associated with the item ( % = /5)    |
|             | Start Date             | Char(14)   |                  | Start date for the new tax rate. (in YYYYMMDD format) |
| TTAIL       | File Record Descriptor | Char(5)    | TTAIL            | Identifies transaction trailer.                       |
|             | Line Number            | Number(10) |                  | Sequential line number                                |
|             | Transaction Number     | Number(10) |                  | Transaction Number                                    |
|             | Detail Counter         | Number(10) |                  | Number of detail records in transaction.              |
| FTAIL       | File Record descriptor | Char(5)    | FTAIL            | Identifies file trailer                               |



| Record Name | Field Name          | Field Type | Default Value | Description                               |
|-------------|---------------------|------------|---------------|-------------------------------------------|
|             | Line Number         | Number(10) |               | Sequential Line Number                    |
|             | Detail Line Counter | Number(10) |               | Total Number of detail records processed. |

## txrposdn (Tax Rate POS Download)

### Functional Area

Tax rate batch

### Module Affected

TXRPOSDN.PC

### Design Overview

This batch program processes records from the GEOCODE\_TXCDE (GEOCODE tax code), PRODUCT\_TAX\_CODE (PRODUCT tax code), and TAX\_RATES (tax rate) tables and writes all item/location combinations to the TIF\_EXPLODE table. A separate batch program TIFPOSDN.PC processes records on the TIF\_EXPLODE table, computes the tax rate and writes it to the POS flat file.

### Scheduling Constraints

| Schedule Information      | Description                         |
|---------------------------|-------------------------------------|
| Processing Cycle          | Phase 4 (daily)                     |
| Scheduling Considerations | Run this module before TIFPOSDN.PC. |
| Pre-Processing            | N/A                                 |
| Post-Processing           | N/A                                 |
| Threading Scheme          | N/A                                 |

### Restart/Recovery

The logical unit of work for the TXRPOSDN module is store/item or store/department with a recommended commit counter setting of 10,000. Each time the record counter equals the maximum recommended commit number, the processed value is stored into the database and saved.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

**Key Tables Affected**

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| ITEM_LOC         | Yes    | No     | No     | No     |
| ITEM_MASTER      | Yes    | No     | No     | No     |
| PRODUCT_TAX_CODE | Yes    | No     | Yes    | No     |
| STORE            | Yes    | No     | No     | No     |
| GEOCODE_TXCDE    | Yes    | No     | Yes    | No     |
| GEOCODE_STORE    | Yes    | No     | No     | No     |
| TAX_RATES        | Yes    | No     | Yes    | No     |
| TIF_EXPLODE      | Yes    | Yes    | No     | No     |
| PERIOD           | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**txrtupld (Tax Rate Upload)****Functional Area**

Tax Rate

**Module Affected**

TXRTUPLD.PC

**Design Overview**

This program is used to upload tax rates from an outside source into the RMS tables. The upload stores provide the means to create new tax jurisdictions, tax codes and tax rates. In addition to the ability to insert new tax codes and rates it stores also be able to update the rates for existing tax codes. The input file contains the tax jurisdiction, tax type, level and tax rates, which stores allow the batch programs to correctly update the tax rates to reflect any changes that have occurred.

**Scheduling Constraints**

| Schedule Information      | Description                  |
|---------------------------|------------------------------|
| Processing Cycle          | Phase 4 (Daily) or as needed |
| Scheduling Considerations | N/A                          |
| Pre-Processing            | N/A                          |
| Post-Processing           | N/A                          |
| Threading Scheme          | N/A                          |

## Restart/Recovery

This program reads the FDETL records one by one, validates them and then loads it to different insert or update arrays. When the commit point is reached, all the arrays are written into the database. The LUW (Logical Unit of Work) is each detailed record in the upload file. The recommended commit max counter is 1000 (this depends on the implementation).

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| TAX_JURISDICTIONS | Yes    | Yes    | Yes    | No     |
| TAX_CODES         | Yes    | Yes    | No     | No     |
| TAX_RATES         | Yes    | Yes    | Yes    | No     |

## I/O Specification

### Input File Layout

| Record Name | Field Name             | Field Type | Default Value | Description                               |
|-------------|------------------------|------------|---------------|-------------------------------------------|
| FHEAD       | File head descriptor   | Char(5)    | FHEAD         | Describes file line type                  |
|             | Line id                | Number(10) | 0000000001    | Sequential file line number               |
|             | Gentran ID             | Char(4)    | 'TXUP'        | Identifies which translation Gentran uses |
|             | Current date           | Char(14)   |               | File date in YYYYMMDDHH24MISS format      |
| FDETL       | File record descriptor | Char(5)    | THEAD         | Start of an invoice transaction           |
|             | Line id                | Number(10) |               | Sequential file line number               |
|             | Tax Jurisdiction       | Number(8)  |               | Tax Jurisdiction                          |

| Record Name | Field Name                   | Field Type  | Default Value | Description                                          |
|-------------|------------------------------|-------------|---------------|------------------------------------------------------|
|             | Tax Jurisdiction Description | Char(120)   |               | Tax Jurisdiction description                         |
|             | Tax Level                    | Char(6)     |               | Tax level of the Jurisdiction                        |
|             | Tax Type                     | Char(6)     |               | Tax Type                                             |
|             | Tax Rate                     | Number(8,5) |               | Tax rate of the tax code                             |
|             | Start Date                   | Char (14)   |               | Start Date of the tax rate                           |
|             | End Date                     | Char (14)   |               | End Date of the tax rate                             |
| FTAIL       | File record descriptor       | Char(5)     | FTAIL         | Marks end of file                                    |
|             | Line id                      | Number(10)  |               | Sequential file line number                          |
|             | Number of lines              | Number(10)  |               | Number of lines in file not counting FHEAD and FTAIL |

---

## Tickets and Labels Batch

### Overview

The tickets and labels batch module, TCKTDNLD.PC, outputs an interface file for an external ticket printing system. The module runs to create an output file containing all information to be printed on a ticket or label for a particular item and location. Ticket attributes can include two types of information:

- **Attributes:** System-defined characteristics of an item. For example, a retailer can specify that the department, class, subclass, and retail price be printed on tickets.
- **User Defined Attributes:** User-defined characteristics of an item. For example, a retailer indicate that a user-defined date, free-form text, or value be printed on tickets.

### Batch Design Summary

The following batch design is included in this functional area:

- TCKTDNLD.PC (Ticket download)

### tcktdnld (Ticket Download)

#### Functional Area

Tickets and Labels

#### Module Affected

TCKTDNLD.PC

#### Design Overview

The TCKTDNLD.PC program creates an output file containing all of the information to be printed on a ticket or label for a particular item/location. This program is driven by the “requests” for tickets that exist on the TICKET\_REQUEST table. Information to be printed on the ticket is then retrieved based on the item, location and the ticket type requested. The details, which should be printed on each type of ticket, are kept on the TICKET\_TYPE\_DETAIL table. Specific details, which are written to the output file, are taken from the various item tables.

For more information, see the chapter “Location Retail Batch” in this volume of the RMS Operations Guide.

#### Scheduling Constraints

| Schedule Information      | Description         |
|---------------------------|---------------------|
| Processing Cycle          | Ad Hoc – Runs Daily |
| Scheduling Considerations | N/A                 |
| Pre-Processing            | N/A                 |
| Post-Processing           | N/A                 |

| Schedule Information | Description |
|----------------------|-------------|
| Threading Scheme     | N/A         |

### Restart/Recovery

Restart ability exists implicitly within this program, because records are deleted after they are selected. Due to the low volume of this program, multi-threading is not used in this program.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table                 | Select | Insert | Update | Delete |
|-----------------------|--------|--------|--------|--------|
| TICKET_REQUEST        | Yes    | No     | No     | Yes    |
| STORE                 | Yes    | No     | No     | No     |
| TICKET_TYPE_HEAD      | Yes    | No     | No     | No     |
| ITEM_MASTER           | Yes    | No     | No     | No     |
| TICKET_TYPE_DETAIL    | Yes    | No     | No     | No     |
| UDA_VALUES            | Yes    | No     | No     | No     |
| UDA_ITEM_LOV          | Yes    | No     | No     | No     |
| UDA                   | Yes    | No     | No     | No     |
| TL_SHADOW             | Yes    | No     | No     | No     |
| UDA_ITEM_FF           | Yes    | No     | No     | No     |
| UDA_ITEM_DATE         | Yes    | No     | No     | No     |
| ITEM_LOC              | Yes    | No     | No     | No     |
| ITEM_SUPPLIER         | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY_DIM | Yes    | No     | No     | No     |
| ITEM_SUPP_COUNTRY     | Yes    | No     | No     | No     |
| DEPS                  | Yes    | No     | No     | No     |
| CLASS                 | Yes    | No     | No     | No     |
| SUBCLASS              | Yes    | No     | No     | No     |
| ORDHEAD               | Yes    | No     | No     | No     |

| Table                  | Select | Insert | Update | Delete |
|------------------------|--------|--------|--------|--------|
| ORDSKU                 | Yes    | No     | No     | No     |
| ITEM_ZONE_PRICE        | Yes    | No     | No     | No     |
| PRICE_ZONE_GROUP_STORE | Yes    | No     | No     | No     |
| WH                     | Yes    | No     | No     | No     |
| VAT_ITEM               | Yes    | No     | No     | No     |

## I/O Specification

### Output File Layout

| Record Name | Field Name                  | Field Type   | Default Value              | Description                                                            |
|-------------|-----------------------------|--------------|----------------------------|------------------------------------------------------------------------|
| FHEAD       | File Type Record Descriptor | Char(5)      | FHEAD                      | Identifies file record type                                            |
|             | File Line Sequence          | Number(10)   |                            | Line number of the current file                                        |
|             | File Type Definition        | Char(4)      | TCKT                       | Identifies file as 'Print Ticket Requests'                             |
|             | File Create Date            | Char(14)     | create date                | Vdate in 'YYMMDDHHMISS' format                                         |
| THEAD       | File Type Record Descriptor | Char(5)      | THEAD                      | Identifies file record type                                            |
|             | File Line Sequence          | Number(10)   |                            | Line number of the current file                                        |
|             | ITEM                        | Char(25)     |                            | ID number of the transaction level, non-pack item or the pack item     |
|             | Ticket Type                 | Char(4)      |                            | ID which indicates the ticket type to be printed                       |
|             | Location Type               | Char(1)      | S - Store<br>W - Warehouse | Identifies the type of location for which tickets are printed          |
|             | Location                    | Char(10)     |                            | number of the store or warehouse for which tickets are printed         |
|             | Quantity                    | Number(12,4) |                            | the quantity of tickets to be printed*10000 (4 implied decimal places) |

| Record Name | Field Name                    | Field Type   | Default Value       | Description                                                                                                                                                                    |
|-------------|-------------------------------|--------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TCOMP       | File Type Record Descriptor   | Char(5)      | TCOMP               | Identifies file record type                                                                                                                                                    |
|             | File Line Sequence            | Number(10)   |                     | Line number of the current file                                                                                                                                                |
|             | ITEM                          | Char(25)     |                     | ID number of the ITEM                                                                                                                                                          |
|             | Quantity                      | Number(12,4) |                     | Quantity of the component ITEM as part of the whole; if ITEM on the header record is a transaction level ITEM, the value in this field is 1. *10000 (4 implied decimal places) |
| TDETL       | File Type Record Descriptor   | Char(5)      | TDETL               | Identifies file record type                                                                                                                                                    |
|             | File Line Sequence            | Number(10)   |                     | Line number of the current file                                                                                                                                                |
|             | Detail Sequence Number        | Number(10)   |                     | Sequential number assigned to the detail records                                                                                                                               |
|             | Ticket Item                   | Char(4)      |                     | ID indicating the detail to be printed on the ticket                                                                                                                           |
|             | Attribute Description         | Char(120)    |                     | Description of the attribute (from the UDA Table)                                                                                                                              |
|             | Value                         | Char(250)    |                     | Detail to be printed on the ticket (that is, REF_ITEM, Department Number, ITEM description)                                                                                    |
|             | Supplement                    | Char(120)    |                     | Supplemental description to the Value (that is, Department Name)                                                                                                               |
| TTAIL       | File Type Record Descriptor   | Char(5)      | TTAIL               | Identifies file record type                                                                                                                                                    |
|             | File Line Sequence            | Number(10)   |                     | Line number of the current file                                                                                                                                                |
|             | Transaction Detail Line Count | Number(6)    | sum of detail lines | Sum of the detail lines within a transaction                                                                                                                                   |



| Record Name | Field Name                  | Field Type | Default Value | Description                     |
|-------------|-----------------------------|------------|---------------|---------------------------------|
| FTAIL       | File Type Record Descriptor | Char(5)    | FTAIL         | Identifies file record type     |
|             | File Line Sequence          | Number(10) |               | Line number of the current file |

---

## Transfers, Allocation, and RTV Batch

### Overview

#### Transfers

A transfer is a movement of stock on hand from one stockholding location within the company to another. The following types of transfers are used in RMS:

- **Administrative:** Stock is transferred for administrative purposes rather than merchandising.
- **Book transfer:** Stock is transferred between two virtual warehouses within the same physical warehouse. The transfer is created, approved, and closed in one step. No shipment records are created. This option is available in a multi-channel environment.
- **Combined transfer:** The system automatically combines store requisition and cross-dock PO transfers that have the same origin and destination locations into one transfer. The store requisition and cross-dock PO transfers are deleted after they are merged into the combined transfer.
- **Confirmation:** The details of the transfer are entered after the transfer has already occurred.
- **Cross-dock PO:** A transfer is created automatically when stock is received with the following characteristics: the items are part of a purchase order that is already allocated, cross-docked, and not pre-marked.
- **Customer order:** Stock is reserved for a customer. The stock may be shipped to the customer's address or held for pickup by the customer.
- **Intercompany:** Stock is transferred between two transfer entities. The transfer may have three locations associated with it: an origin location, a finishing location, and a destination location, or two locations associated with it: an origin location and a destination location. The finishing location on an intercompany transfer with finishing must have the same transfer entity as either the origin location or the destination location.
- **Manual requisition:** A manual requisition is used as a general purpose transfer when no other type of transfer is applicable. An example might be a store to store transfer.
- **Non-salable book transfer:** Stock that is marked as non-salable is moved from one virtual warehouse's unavailable inventory area to another virtual warehouse's unavailable inventory area within the same physical warehouse. The transfer is created, approved, and closed in one step. No shipment records are created. This option is available in a multi-channel environment.
- **Non-salable merchandise:** Stock that is marked as non-salable is moved from one unavailable inventory area to another, such as to a repair center.
- **PO-linked transfer:** When not enough stock is available at a warehouse in order to fill a store order, the transfer is linked to a purchase order that was created in order to satisfy the remaining need. The transfer is created automatically by the system.

- **Reallocation transfer:** A reallocation transfer allows a retailer to ship across legal entities without restriction. For example, it allows a retailer to move stock from stores to warehouses, so that warehouses can reallocate that stock.
- **Return to vendor:** Stock that is marked as a return to vendor is transferred to a consolidation location.
- **Store requisition:** Stock is transferred based on replenishment needs. The transfer is created automatically by the system.

Store requisition, cross-dock PO, PO-linked, and combined transfers are created automatically. All other types of transfers are created manually for a variety of purposes.

If you have access to a warehouse management system (WMS) such as Oracle Retail Warehouse Management System (RWMS), transfer, shipment, and receipt details can be transmitted between the two systems.

## Returns to Vendor (RTVs)

A return to vendor (RTV) order is used to send merchandise back to the supplier. One or more items may be included on the RTV order, but only one supplier and location can be entered.

RTV orders may be received from an external system, such as RWMS. The items on these RTV orders are already shipped, so their status in RMS is 'Shipped'.

RTVs are created by using the last receipt cost. If the last receipt cost cannot be found, then the Weighted Average Cost (WAC) is used. The last receipt cost is the cost of an item the last time a retailer purchased it from a supplier.

## Mass Return Transfers (MRT)

Return transfers are transfers from stores to warehouses. Generally, return transfers are made for one of the two following reasons:

- to redistribute merchandise from one store to other locations
- to return merchandise to the vendor

Mass return transfers (MRTs) from several locations to single warehouses are similar to item allocations. However, where an item allocation distributes items from one warehouse to many stores, an MRT returns items from several locations to one warehouse. After the items are returned to the warehouse, the retailer can return them to the supplier.

This automates the process of manual creation of:

- Transferring stock from multiple locations to one receiving warehouse
- Associating RTVs
- Facilitating the management of all related transfers and RTVs through one dialog

## Batch Design Summary

The following batch designs are included in this functional area:

- DISTROPCPUB.PC (Distro Price Change Publish)
- DOCCLOSE.PC (Document Close)
- MRT.PC (Mass Transfer Creation)
- MRTPRG (Mass Return Transfer Purge)
- MRTRTV.PC (Mass RTV Creation)

- MRTUPD.PC (Mass Return Update)
- RTVPRG (Return to Vendor Purge)
- TSFPRG.PC (Transfer Purge)

## distrocpub (Distro Price Change Publish)

### Functional Area

Pricing/Transfers/Allocations

### Module Affected

DISTROPCPUB.PC

### Design Overview

The DISTROPCPUB.PC program stores get price change information for any allocations and transfers and write the information to the corresponding queue table. This program stores ensure that Oracle Retail Warehouse Management has access to any item/location unit retail information that is changed after an allocation or transfer is published.

This program stores loop through the PRICE\_HIST table, selecting records whose unit retail stores change for vdate+1, and transaction type is in 4 (single unit retail was changed) or 11(single unit retail and multi-unit retail were changed). It stores then search for allocations and transfers with matching item/locations. When a match is found, depending on the distro type, the program calls allocation or transfer publishing logic to insert the data into the allocation or transfer queue table, so that the RIB can publish the change to the warehouse system.

### Scheduling Constraints

| Schedule Information      | Description                                                            |
|---------------------------|------------------------------------------------------------------------|
| Processing Cycle          | Phase 3 – Daily                                                        |
| Scheduling Considerations | This program should run after RPM price event execution batch process. |
| Pre-Processing            | N/A                                                                    |
| Post-Processing           | N/A                                                                    |
| Threading Scheme          | Multithreading based on store.                                         |

### Restart/Recovery

The logical unit of work is store. The driving cursor retrieves all item/locations that have price changes in effect from the next day. It also gets all of the component items of the non-sellable packs that have price changes.

### Locking Strategy

N/A

### Security Considerations

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| PERIOD          | Yes    | No     | No     | No     |
| PRICE_HIST      | Yes    | No     | No     | No     |
| V_RESTART_STORE | Yes    | No     | No     | No     |
| V_PACKSKU_QTY   | Yes    | No     | No     | No     |
| ITEM_MASTER     | Yes    | No     | No     | No     |
| ALLOC_HEADER    | Yes    | No     | No     | No     |
| ALLOC_DETAIL    | Yes    | No     | No     | No     |
| TSFHEAD         | Yes    | No     | No     | No     |
| TSFDETAIL       | Yes    | No     | No     | No     |
| ORDHEAD_REV     | Yes    | No     | No     | No     |
| ORDHEAD         | Yes    | No     | No     | No     |
| ALLOC_MFQUEUE   | No     | Yes    | No     | No     |
| TSF_MFQUEUE     | No     | Yes    | No     | No     |

**Shared Modules**

N/A

**docclose (Document Close)****Functional Area**

Appointments

**Module Affected**

DOCCLOSE.PC

**Design Overview**

This program is used to close un-appointed receipts (POs, transfers or allocations). These receipts do not have an associated appointment record within RMS (and therefore cannot be closed via appointment processing).

Un-appointed receipts are recorded on the DOC\_CLOSE\_QUEUE table. This batch program stores retrieve unique documents from the the table and use existing functions to attempt closure for each. all entries for any closed document stores then be deleted from the table.

**Scheduling Constraints**

| Schedule Information      | Description    |
|---------------------------|----------------|
| Processing Cycle          | Ad hoc (daily) |
| Scheduling Considerations | N/A            |
| Pre-Processing            | N/A            |
| Post-Processing           | N/A            |
| Threading Scheme          | N/A            |

**Restart/Recovery**

The logical unit of work is a unique doc and doc\_type combination. The program is restartable on the doc number.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table           | Select | Insert | Update | Delete |
|-----------------|--------|--------|--------|--------|
| DOC_CLOSE_QUEUE | Yes    | No     | No     | Yes    |
| ORDHEAD         | No     | No     | Yes    | No     |
| DEAL_CALC_QUEUE | No     | No     | No     | Yes    |
| ITEM_LOC_SOH    | No     | No     | Yes    | No     |
| TSFHEAD         | No     | No     | Yes    | No     |
| ALLOC_HEADER    | No     | No     | Yes    | No     |

**Shared Modules**

N/A

**I/O Specification**

N/A

**mrt (Mass Return Transfer)****Functional Area**

Return to vendor.

## Module Affected

MRT.PC

## Design Overview

The MRT.PC batch program creates individual transfers for each from location for all approved Mass Return Transfers created using the MRT form. Transfers that are generated has a default status of 'A' (Approved). However for MRTs with a Quantity Type of "Manual", if the SOH at the sending location is lower than the requested quantity the status is set to 'I' (Input). In addition, if the MRT is in approved status but the NOT\_AFTER\_DATE is later than the vdate, the status of the associated transfers stores also be set to 'I' (Input).

## Scheduling Constraints

| Schedule Information      | Description                                                                                                    |
|---------------------------|----------------------------------------------------------------------------------------------------------------|
| Processing Cycle          | Phase 2                                                                                                        |
| Scheduling Considerations | This batch should be scheduled to run before MRTUPD and MRTRTV, and before any other transfer-related batches. |
| Pre-Processing            | N/A                                                                                                            |
| Post-Processing           | N/A                                                                                                            |
| Threading Scheme          | Threaded by warehouse                                                                                          |

## Restart/Recovery

The logical unit of work is a from location/to location combination. Multiple items comprise a LUW. This represents a transfer of multiple items from a location (store or warehouse) to a warehouse. Restart/recovery is based on from location/to location as well.

The batch program uses the v\_restart\_all\_locations view to thread processing by warehouse (to location)

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table    | Select | Insert | Update | Delete |
|----------|--------|--------|--------|--------|
| MRT      | Yes    | No     | Yes    | No     |
| MRT_ITEM | Yes    | No     | No     | No     |

| Table             | Select | Insert | Update | Delete |
|-------------------|--------|--------|--------|--------|
| MRT_ITEM_LOC      | Yes    | No     | Yes    | No     |
| ITEM_SUPP_COUNTRY | Yes    | No     | No     | No     |
| ITEM_LOC          | Yes    | No     | No     | No     |
| ITEM_MASTER       | Yes    | No     | No     | No     |
| ITEM_LOC_SOH      | Yes    | No     | Yes    | No     |
| TSFDETAIL         | Yes    | Yes    | Yes    | No     |
| TSFHEAD           | No     | Yes    | Yes    | No     |
| TSF_ITEM_COST     | No     | Yes    | No     | No     |
| TRAN_DATA         | No     | Yes    | No     | No     |
| INV_STATUS_QTY    | Yes    | Yes    | Yes    | Yes    |
| PERIOD            | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS    | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**mrtprg (Mass Return Transfer Purge)****Functional Area**

Return to vendor.

**Module Affected**

MRTPRG.PC

**Design Overview**

The purpose of this module is to purge mass return transfer (MRT) records, and their associated transfers and RTVs.

Only MRTs with a status of closed where the close date plus the tsf\_mrt\_retention\_days value specified on SYSTEM\_OPTIONS is earlier than tomorrows date is processed. The RTV records must also not have associated transfer records with a status of closed or deleted.

**Scheduling Constraints**

| Schedule Information      | Description                    |
|---------------------------|--------------------------------|
| Processing Cycle          | AD HOC                         |
| Scheduling Considerations | This program should run daily. |
| Pre-Processing            | N/A                            |
| Post-Processing           | N/A                            |



| Schedule Information | Description            |
|----------------------|------------------------|
| Threading Scheme     | Threaded by warehouse. |

### Restart/Recovery

The logical unit of work for this batch program is a warehouse location. The program is multithreaded using v\_restart\_all\_locations view.

### Locking Strategy

The following tables are locked by the package INVC\_SQL for update:

- INVC\_DETAIL
- INVC\_NON\_MERCH
- INVC\_MERCH\_VAT
- INVC\_DETAIL\_VAT
- INVC\_MATCH\_QUEUE
- INVC\_HEAD
- INVC\_DISCOUNT
- INVC\_TOLERANCE
- ORDLOC\_INVC\_COST
- INVC\_MATCH\_WKSHT
- INVC\_XREF

The following tables are locked by the program directly for update:

- MRT\_ITEM\_LOC
- MRT\_ITEM
- MRT

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| PERIOD         | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |
| TSFHEAD        | Yes    | No     | No     | Yes    |
| TSFDETAIL      | No     | No     | No     | Yes    |
| SHIPMENT       | No     | No     | No     | Yes    |
| SHIPSKU        | Yes    | No     | No     | Yes    |
| ORDCUST        | Yes    | No     | No     | Yes    |

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| SHIPITEM_INV_FLOW  | No     | No     | No     | Yes    |
| CARTON             | No     | No     | No     | Yes    |
| APPT_HEAD          | Yes    | No     | No     | Yes    |
| APPT_DETAIL        | Yes    | No     | No     | Yes    |
| DOC_CLOSE_QUEUE    | No     | No     | No     | Yes    |
| INVC_HEAD          | Yes    | No     | No     | Yes    |
| INVC_DETAIL        | Yes    | No     | No     | Yes    |
| INVC_DETAIL_VAT    | Yes    | No     | No     | Yes    |
| INVC_NON_MERCH     | Yes    | No     | No     | Yes    |
| INVC_MERCH_VAT     | Yes    | No     | No     | Yes    |
| INVC_DISCOUNT      | Yes    | No     | No     | Yes    |
| INVC_TOLERANCE     | Yes    | No     | No     | Yes    |
| INVC_MATCH_QUEUE   | Yes    | No     | No     | Yes    |
| INVC_MATCH_WKSHT   | Yes    | No     | No     | Yes    |
| INVC_XREF          | Yes    | No     | No     | Yes    |
| ORDLOC_INVC_COST   | Yes    | No     | Yes    | No     |
| MRT                | Yes    | No     | No     | Yes    |
| MRT_ITEM           | Yes    | No     | No     | Yes    |
| MRT_ITEM_LOC       | Yes    | No     | No     | Yes    |
| RTV_HEAD           | Yes    | No     | No     | Yes    |
| RTV_DETAIL         | No     | No     | No     | Yes    |
| TSFDETAIL_CHRG     | No     | No     | No     | Yes    |
| TSF_WO_DETAIL      | No     | No     | No     | Yes    |
| TSF_WO_HEAD        | No     | No     | No     | Yes    |
| TSF_XFORM_DETAIL   | No     | No     | No     | Yes    |
| TSF_XFORM          | No     | No     | No     | Yes    |
| TSF_PACKING_DETAIL | No     | No     | No     | Yes    |
| TSF_PACKING        | No     | No     | No     | Yes    |
| TSF_ITEM_WO_COST   | No     | No     | No     | Yes    |
| TSF_ITEM_COST      | No     | No     | No     | Yes    |

**I/O Specification**

N/A

## mrrtrtv (Mass Return to Vendor Creation)

### Functional Area

Transfers and Returns to Vendor

### Module Affected

MRTRTV.PC

### Design Overview

This batch program creates RTVs for mass return transfers that require an RTV to be created automatically. This program creates RTVs for MRTs that have an `rtv_create_date` earlier than or equal to the current date. It reads data from the `mrt` tables and constructs `RTV_HEAD` and `RTV_DETAIL` information. The program stores then set the status of all processed MRTs to 'R' in the `MRT` table.

### Scheduling Constraints

| Schedule Information      | Description                                           |
|---------------------------|-------------------------------------------------------|
| Processing Cycle          | Phase 2 (Daily)                                       |
| Scheduling Considerations | Before <code>mrtupd</code> and after <code>mrt</code> |
| Pre-Processing            | N/A                                                   |
| Post-Processing           | N/A                                                   |
| Threading Scheme          | Threaded by warehouse                                 |

### Restart/Recovery

The logical unit of work for this program is set at the warehouse level. Threading is done by store using the `v_restart_all_locations` view.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table        | Select | Insert | Update | Delete |
|--------------|--------|--------|--------|--------|
| PERIOD       | Yes    | No     | No     | No     |
| MRT          | Yes    | No     | Yes    | No     |
| MRT_ITEM     | Yes    | No     | No     | No     |
| MRT_ITEM_LOC | Yes    | No     | No     | No     |

| Table      | Select | Insert | Update | Delete |
|------------|--------|--------|--------|--------|
| SUPS       | Yes    | No     | No     | No     |
| RTV_HEAD   | No     | Yes    | Yes    | No     |
| RTV_DETAIL | No     | Yes    | No     | No     |
| ADDR       | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**mrtupd (Mass Return Transfer Update)****Functional Area**

Transfers &amp; Returns to Vendor

**Module Affected**

MRTUPD.PC

**Design Overview**

This program updates the status of MRTs and their associated transfers.

**Scheduling Constraints**

| Schedule Information      | Description           |
|---------------------------|-----------------------|
| Processing Cycle          | Phase 2               |
| Scheduling Considerations | Run after mrtrtv.pc   |
| Pre-Processing            | N/A                   |
| Post-Processing           | N/A                   |
| Threading Scheme          | Threaded by warehouse |

**Restart/Recovery**

The logical unit of work for this program is wh. This program is multithreaded using the v\_restart\_all\_locations view.

**Locking Strategy**

N/A

**Security Considerations**

N/A

**Performance Considerations**

N/A

**Key Tables Affected**

| Table         | Select | Insert | Update | Delete |
|---------------|--------|--------|--------|--------|
| PERIOD        | Yes    | No     | No     | No     |
| MRT           | Yes    | No     | Yes    | No     |
| TSFHEAD       | Yes    | No     | Yes    | No     |
| SHIPSKU       | Yes    | No     | No     | No     |
| TSFDETAIL     | Yes    | No     | No     | No     |
| ITEM_MASTER   | Yes    | No     | No     | No     |
| ITEM_LOC      | Yes    | No     | No     | No     |
| ITEM_LOC_SOH  | Yes    | No     | Yes    | No     |
| V_PACKSKU_QTY | Yes    | No     | No     | No     |

**I/O Specification**

N/A

**rtvprg (Return to Vendor Purge)****Functional Area**

RTV

**Module Affected**

RTVPRG.PC

**Design Overview**

This batch program purges outdated RTV transactions from RMS. RTV transactions that are completely shipped or cancelled or those for which the completion date exceeds the RTV retention period is purged. RTV's with a debit memo not in 'P'osted status (present in INVC\_HEAD) stores not be purged. The RTV retention period is maintained as a system level variable in the unit\_options table.

**Scheduling Constraints**

| Schedule Information      | Description      |
|---------------------------|------------------|
| Processing Cycle          | AD HOC (monthly) |
| Scheduling Considerations | N/A              |
| Pre-Processing            | N/A              |
| Post-Processing           | N/A              |
| Threading Scheme          | N/A              |

**Restart/Recovery**

No restart recovery logic.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table            | Select | Insert | Update | Delete |
|------------------|--------|--------|--------|--------|
| RTV_HEAD         | No     | No     | No     | Yes    |
| RTV_DETAIL       | No     | No     | No     | Yes    |
| INVC_HEAD        | Yes    | No     | No     | Yes    |
| INVC_DETAIL      | No     | No     | No     | Yes    |
| INVC_NON_MERCH   | Yes    | No     | No     | Yes    |
| INVC_MERCH_VAT   | Yes    | No     | No     | Yes    |
| INVC_DETAIL_VAT  | Yes    | No     | No     | Yes    |
| INVC_MATCH_QUEUE | Yes    | No     | No     | Yes    |
| INVC_DISCOUNT    | Yes    | No     | No     | Yes    |
| INVC_TOLERANCE   | Yes    | No     | No     | Yes    |
| ORDLOC_INVC_COST | Yes    | No     | Yes    | No     |
| INVC_MATCH_WKSHT | Yes    | No     | No     | Yes    |
| INVC_XREF        | Yes    | No     | No     | Yes    |

### I/O Specification

NA

## tsfprg (Transfer Purge)

### Functional Area

Transfers

### Module Affected

TSFPRG.PC

## Design Overview

This module purges closed or deleted transfers after a set number of days. The number of days to retain closed and deleted transfers is set in the SYSTEM\_OPTIONS table. If a transfer has allocations associated to it, all these allocations and associated tier records must be closed first before the transfer records can be purged. This batch program does not process Mass Return Transfers (MRT). Purging of MRT records are done by mrtprg.pc.

## Scheduling Constraints

| Schedule Information      | Description                 |
|---------------------------|-----------------------------|
| Processing Cycle          | AD-HOC (daily)              |
| Scheduling Considerations | N/A                         |
| Pre-Processing            | N/A                         |
| Post-Processing           | N/A                         |
| Threading Scheme          | Threaded by transfer number |

## Restart/Recovery

This batch program is multithreaded using the v\_restart\_transfer view. The logical unit of work is a transfer number. This batch program commits to the database for every commit\_max\_ctr number of transfers processed.

## Locking Strategy

N/A

## Security Considerations

N/A

## Performance Considerations

N/A

## Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| PERIOD         | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |
| TSFHEAD        | Yes    | No     | No     | Yes    |
| TSFDETAIL      | No     | No     | No     | Yes    |
| TSFDETAIL_CHRG | No     | No     | No     | Yes    |
| TSF_WO_HEAD    | No     | No     | No     | Yes    |
| TSF_WO_DETAIL  | No     | No     | No     | Yes    |
| TSF_XFORM      | No     | No     | No     | Yes    |

| Table              | Select | Insert | Update | Delete |
|--------------------|--------|--------|--------|--------|
| TSF_XFORM_DETAIL   | No     | No     | No     | Yes    |
| TSF_PACKING        | No     | No     | No     | Yes    |
| TSF_PACKING_DETAIL | No     | No     | No     | Yes    |
| TSF_ITEM_WO_COST   | No     | No     | No     | Yes    |
| TSF_ITEM_COST      | No     | No     | No     | Yes    |
| ALLOC_HEADER       | Yes    | No     | No     | Yes    |
| ALLOC_CHRG         | No     | No     | No     | Yes    |
| ALLOC_DETAIL       | No     | No     | No     | Yes    |
| SHIPSKU            | Yes    | No     | No     | Yes    |
| SHIPMENT           | No     | No     | No     | Yes    |
| CARTON             | No     | No     | No     | Yes    |
| SHIPITEM_INV_FLOW  | No     | No     | No     | Yes    |
| APPT_HEAD          | Yes    | No     | No     | Yes    |
| APPT_DETAIL        | Yes    | No     | No     | Yes    |
| DOC_CLOSE_QUEUE    | No     | No     | No     | Yes    |
| ORDCUST            | No     | No     | No     | Yes    |

**I/O Specification**

N/A



---

## Value Added Tax (VAT) Maintenance Batch

### Overview

Value-added tax (VAT) functionality is optional in RMS. In several countries, value added taxes must be considered when determining the monetary value of items. VAT amounts appear in several modules of the system, such as purchase orders, pricing, contracts, stock ledger, and invoice matching. This overview describes the RMS system settings that impact VAT, along with the batch module VATDLXPL.PC that associates items with a given VAT region and VAT code.

Value added tax rates are identified by VAT code. When VAT codes are associated with a VAT region, they are assigned a VAT type. The VAT type indicates that the tax rate is used in one of the following types of calculations:

- Cost: The tax rate is applied to purchase transactions.
- Retail: The tax rate is applied to sales transactions.
- Both: The tax rate is applied to purchase and sales transactions.

Value added taxes are reflected in the stock ledger when 1) the retail method of accounting is used and 2) the system is set up to include VAT in retail calculations.

A number of the system settings in RMS, which are described beginning in the next section, indicate how you wish to implement VAT.

### System Level VAT

The `vat_ind` column on the `SYSTEM_OPTIONS` table is the primary means to initiate VAT in RMS. If the value in this column is 'Y', then RMS stores include VAT in the system.

### System Class Level VAT

The `class_level_vat_ind` column on the `SYSTEM_OPTIONS` table allows you to include or exclude VAT at the class level of the merchandise hierarchy. A value of 'Y' in this column allows you to manage VAT inclusion or exclusion from retail at the class level. A value of 'N' means that VAT is included in the retail price in RMS and in the point-of-sale (POS) download for all classes. The POS upload process is controlled by the store VAT indicator, which is described later in this overview.

### Department VAT

The department table (DEPS) holds the `dept_vat_incl_ind` column that is used to enable or disable VAT in retail prices for all classes in the department. This indicator is used only to default to the class level indicator when classes are initially set up for the department and is only available when the system level class VAT option is on. When VAT is turned on in the system and not defined at the class level, this field defaults to 'Y'. When VAT is turned off in the system, this field defaults to 'N'.

## Class VAT

The `class_vat_ind` column on the `CLASS` table determines if retail is displayed and held with or without VAT. The default setting is inherited from the class's department. You can edit the value in this column only when VAT is turned on in the system and defined at the class level.

If the value in this column is 'Y', VAT is included in the retail price for all items in that class. Both point-of-sale (POS) download (`POSDNLD.PC`) and POS upload (`POSUPLD.PC`) stores include VAT in the retail price.

If the value in this column is 'N', VAT is excluded from POS download (`POSDNLD.PC`) and POS upload (`POSUPLD.PC`) of retail prices for the entire class.

Instructions that are sent to allow the POS to add VAT are contained in these columns on the `POS_MODS` table:

- `Vat_code` – code for the VAT rate
- `Vat_rate` – the actual rate referenced by the VAT code
- `Class_vat_ind`

## Store VAT Indicator

If the value in the `class_level_vat_ind` column on the `SYSTEM_OPTIONS` table is 'N', you can still choose VAT settings for a store. The `vat_include_ind` column on the `STORE` table allows you to include or exclude VAT at the store for POS upload only.

If the value in this column is 'Y', VAT stores always be included in the retail price in the POS upload process. If the value in this column is 'N', VAT stores always be excluded from POS uploaded prices.

## Send VAT Rate to POS

VAT rates are sent through the POS to the store and are contained in these columns on the `POS_MODS` table:

- `Vat_code` – code for the VAT rate
- `Vat_rate` – the actual rate referenced by the VAT code
- `Class_vat_ind`

## Special Note: Retail Method Stock Ledger and VAT

If the stock ledger for a department is set to use the retail method of accounting, an additional setting is required to ensure that VAT is, or is not, included in retail values. If the value in the `STKLDGR_VAT_INCL_RETL_IND` column (`SYSTEM_OPTIONS` table) is 'Y', all retail values for that department in the stock ledger (sales retail, purchase retail, gross margin, and so on) are VAT inclusive. 'N' indicates that VAT is excluded from retail values.

## Batch Design Summary

The following batch design is included in this functional area:

- `VATDLXPL.PC` (VAT Download Explode)

## vatdlxpl (VAT Download Explode)

### Functional Area

Value added tax (VAT) maintenance

### Module Affected

VATDLXPL.PC

### Design Overview

This batch program updates VAT information for each item associated with a given VAT region and VAT code.

### Scheduling Constraints

| Schedule Information      | Description                        |
|---------------------------|------------------------------------|
| Processing Cycle          | Phase 0 (daily)                    |
| Scheduling Considerations | Run as Needed                      |
| Pre-Processing            | N/A                                |
| Post-Processing           | Run prepost vatdlxpl post program. |
| Threading Scheme          | N/A                                |

### Restart/Recovery

This batch program does intermittent commits to the database for every pi\_commit\_max\_ctr number of rows.

### Locking Strategy

N/A

### Security Considerations

N/A

### Performance Considerations

N/A

### Key Tables Affected

| Table          | Select | Insert | Update | Delete |
|----------------|--------|--------|--------|--------|
| PERIOD         | Yes    | No     | No     | No     |
| SYSTEM_OPTIONS | Yes    | No     | No     | No     |
| VAT_CODE_RATES | Yes    | No     | No     | No     |
| POS_MODS       | No     | Yes    | No     | No     |
| VAT_ITEM       | Yes    | Yes    | Yes    | No     |
| ITEM_LOC       | Yes    | No     | No     | No     |

| Table       | Select | Insert | Update | Delete |
|-------------|--------|--------|--------|--------|
| ITEM_MASTER | Yes    | No     | No     | No     |
| STORE       | Yes    | No     | No     | No     |
| CLASS       | Yes    | No     | No     | No     |

**I/O Specification**

N/A