Value-Added Reseller (VAR) Language

Oracle Retail VAR Applications

The following restrictions and provisions only apply to the programs referred to in this section and licensed to you. You acknowledge that the programs may contain third party software (VAR applications) licensed to Oracle. Depending upon your product and its version number, the VAR applications may include:

(i) the software component known as ACUMATE developed and licensed by Lucent Technologies Inc. of Murray Hill, New Jersey, to Oracle and imbedded in the Oracle Retail Predictive Application Server – Enterprise Engine, Oracle Retail Category Management, Oracle Retail Item Planning, Oracle Retail Merchandise Financial Planning, Oracle Retail Advanced Inventory Planning, Oracle Retail Demand Forecasting, Oracle Retail Regular Price Optimization, Oracle Retail Size Profile Optimization, Oracle Retail Replenishment Optimization applications.

(ii) the MicroStrategy Components developed and licensed by MicroStrategy Services Corporation (MicroStrategy) of McLean, Virginia to Oracle and imbedded in the MicroStrategy for Oracle Retail Data Warehouse and MicroStrategy for Oracle Retail Planning & Optimization applications.

(iii) the SeeBeyond component developed and licensed by Sun Microsystems, Inc. (Sun) of Santa Clara, California, to Oracle and imbedded in the Oracle Retail Integration Bus application.

(iv) the Wavelink component developed and licensed by Wavelink Corporation (Wavelink) of Kirkland, Washington, to Oracle and imbedded in Oracle Retail Mobile Store Inventory Management.

(v) the software component known as Crystal Enterprise Professional and/or Crystal Reports Professional licensed by SAP and imbedded in Oracle Retail Store Inventory Management.

(vi) the software component known as Access Via™ licensed by Access Via of Seattle, Washington, and imbedded in Oracle Retail Signs and Oracle Retail Labels and Tags.

(vii) the software component known as Adobe Flex™ licensed by Adobe Systems Incorporated of San Jose, California, and imbedded in Oracle Retail Promotion Planning & Optimization application.

(viii) the software component known as Style Report™ developed and licensed by InetSoft Technology Corp. of Piscataway, New Jersey, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.

(ix) the software component known as DataBeacon™ developed and licensed by Cognos Incorporated of Ottawa, Ontario, Canada, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.

You acknowledge and confirm that Oracle grants you use of only the object code of the VAR Applications. Oracle will not deliver source code to the VAR Applications to you. Notwithstanding any other term or condition of the agreement and this ordering document, you shall not cause or permit alteration of any VAR Applications. For purposes of this section, “alteration” refers to all alterations, translations, upgrades, enhancements, customizations or modifications of all or any portion of the VAR Applications including all reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

The VAR Applications contain trade secrets of Oracle and Oracle’s licensors and Customer shall not attempt, cause, or permit the alteration, decompilation, reverse engineering, disassembly or other reduction of the VAR Applications to a human perceivable form. Oracle reserves the right to replace, with functional equivalent software, any of the VAR Applications in future releases of the applicable program.
# Contents

Send Us Your Comments.............................................................................................................. vii

Preface ............................................................................................................................................ ix
  Audience ..................................................................................................................................... ix
  Related Documents .................................................................................................................. ix
  Customer Support ..................................................................................................................... ix
  Review Patch Documentation ................................................................................................. x
  Oracle Retail Documentation on the Oracle Technology Network ....................................... x
  Conventions ............................................................................................................................... x

1 Introduction to Merchandising Batch Processing ................................................................. 1
  Batch Processing ...................................................................................................................... 1
    Types of Batch Programs .................................................................................................... 1
    Batch Window .................................................................................................................... 2
    Batch Schedule and Phases ............................................................................................... 2
  Merchandising Batch Schedule ............................................................................................. 3
  Program List ............................................................................................................................ 3
  Batch Schedule Diagram ....................................................................................................... 5
    RMS, ReIM, RTM Section ................................................................................................. 5
    ReSA Section ..................................................................................................................... 6
    RPM Section ..................................................................................................................... 6
    Notations in the Batch Schedule Diagram ....................................................................... 7
    prepost Program ................................................................................................................ 8
    Modifications to the Batch Schedule ................................................................................ 9

2 Program List ........................................................................................................................... 11

3 Batch Schedule Diagram ....................................................................................................... 17

4 Interface Diagrams for RMS and RPAS ............................................................................. 19
  RMS Pre/Post Extract Diagrams .......................................................................................... 20
  RMS Foundation Data Extract Diagrams ............................................................................. 21
  RMS Fact Data Extract Diagrams .......................................................................................... 23
  RPAS-RMS Fact Load Diagram ........................................................................................... 24

5 Interface Diagrams for RMS and MFP ................................................................................. 25
  RMS Pre/Post Extract Diagrams .......................................................................................... 26
  RMS Foundation Data Extract Diagrams ............................................................................. 27
  RMS Fact Data Extract Diagrams .......................................................................................... 29

6 Interface Diagrams for RMS and RDW ............................................................................... 31

7 Interface Diagram for RPM and RDW ................................................................................ 43

8 Interface Diagram for ReIM and RDW ................................................................................. 45
9 Interface Diagrams for RMS and AIP ................................................................. 47
   RMS Pre/Post Extract Diagrams ........................................................................ 48
   RMS Foundation Data Extract Diagrams ............................................................. 49
Send Us Your Comments

Oracle Retail Merchandising, Batch Schedule, Release 13.1.2

Oracle welcomes customers’ comments and suggestions on the quality and usefulness of this document. Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

**Note:** Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Applications Release Online Documentation CD available on My Oracle Support and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.
This batch schedule document details the integrated cyclical processing schedules for the Oracle Retail Merchandising applications:

- Oracle Retail Merchandising System (RMS)
- Oracle Retail Invoice Matching (ReIM)
- Oracle Retail Price Management (RPM)
- Oracle Retail Sales Audit (ReSA)
- Oracle Retail Trade Management (RTM)
- Oracle Retail Allocation

**Note:** Although Oracle Retail Allocation is a Merchandising application, it is not represented in this batch schedule because it does not have any batch programs to run. All Allocation processing is online processing.

This guide describes the periodic and ad hoc phases of batch processing, as well as pre- and post-processing dependencies.

**Audience**

The audiences for this guide are as follows:

- Systems analysts and system operations personnel who need information about Merchandising processes, internally or in relation to systems across the enterprise
- Integrators and implementation staff who have the overall responsibility for implementing the Merchandising applications in their enterprise

**Related Documents**

For more information, see the following documents for the Oracle Retail Merchandising products:

- Oracle Retail Data Warehouse Operations Guide
- Oracle Retail Invoice Matching Operations Guide
- Oracle Retail Merchandising System Operations Guide
- Oracle Retail Price Management Operations Guide

**Customer Support**

To contact Oracle Customer Support, access My Oracle Support at the following URL: https://metalink.oracle.com

When contacting Customer Support, please provide the following:

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

If you are installing the application for the first time, you install either a base release (for example, 13.0) or a later patch release (for example, 13.0.2). If you are installing a software version other than the base release, be sure to read the documentation for each patch release (since the base release) before you begin installation. Patch documentation can contain critical information related to the base release and code changes that have been made since the base release.

Oracle Retail Documentation on the Oracle Technology Network

In addition to being packaged with each product release (on the base or patch level), all Oracle Retail documentation is available on the following Web site (with the exception of the Data Model which is only available with the release packaged code):

http://www.oracle.com/technology/documentation/oracle_retail.html

Documentation should be available on this Web site within a month after a product release. Note that documentation is always available with the packaged code on the release date.

Conventions

Navigate: This is a navigate statement. It tells you how to get to the start of the procedure and ends with a screen shot of the starting point and the statement “the Window Name window opens.”

Note: This is a note. It is used to call out information that is important, but not necessarily part of the procedure.

This is a code sample

It is used to display examples of code

A hyperlink appears like this.
Introduction to Merchandising Batch Processing

This chapter is a brief introduction to Oracle Retail batch processing. It defines basic terms and concepts, describes batch processing phases, and explains how to interpret the batch schedule diagram and program list.

**Batch Processing**

Batch processing is the execution of a group of batch programs (jobs). The results are returned without user intervention. Batch programs are commonly used for the following reasons:

- To process large volumes of transaction data
- To interface with external systems
- To perform internal maintenance

Batch programs can process very large quantities of data quickly and efficiently. Batch programs can perform some updates that could be performed through online transactions, but much more quickly and with less impact on system performance. Batch processing is usually scheduled for times when systems are idle or least busy.

Batch programs can be run automatically using batch scheduler software. The batch scheduler allows batch jobs to be set up in a specific order, with restrictions attached to any program as needed. If an error occurs with a batch program, an administrator must correct the error and manually rerun the batch program that failed.

**Types of Batch Programs**

Oracle Retail batch programs are of several types:

- Upload programs bring data from external systems into the Oracle Retail database. For example, the posupld program uploads daily transactions that occur at the point of sale (POS) for processing by the Oracle Retail Management System (RMS).
- Download programs extract data from RMS and format it so it can be used by external systems. For example, the posdnld program extracts new and changed information about an item/location for downloading to the point of sale.
- System maintenance programs perform tasks such as updating the system date. For example, the dtesys program increments the system date at the end of each batch cycle.
- Functional maintenance programs process data specific to a functional area. For example, the storeadd program updates a number of tables to create entries for a new store.
Batch Window

Because of the impact on production systems, it is not always possible to run batch programs during business hours; however, there is a window of opportunity during each day or night when online systems are not being used. This time frame is the batch window. For example, a retailer with stores throughout the continental U.S. might require its online systems to be available from 8 AM Eastern Standard Time, when its East Coast offices open, until 9 PM Pacific Standard Time, when its West Coast stores close. This allows an eight-hour batch window for processing all batch jobs.

Batch Schedule and Phases

Order is critical when running batch programs. Some tasks need to be performed before others. A batch schedule ensures that every time batch processing is performed, the correct tasks are performed in the proper order.

The batch schedule is a diagram that represents all batch programs and how they are sequenced. For each individual user, the schedule is a suggested starting point for the installation. Some programs are specific to products that may not be installed, so these programs may not be used at all.

The total batch schedule is divided into phases. Each phase must be completed before the next phase can begin. Within a phase, there may also be programs that depend on the completion of another program within that phase, so programs within each phase may need to be run in a particular order.
Merchandising Batch Schedule

The integrated Merchandising batch schedule combines the batch schedules of all Merchandising applications into a single schedule diagram. The diagram (later in this document) shows the batch dependencies among the Merchandising applications. The integrated Merchandising batch schedule combines the batch modules for the following applications:

- Oracle Retail Merchandising System (RMS)
- Oracle Retail Trade Management (RTM)
- Oracle Retail Sales Audit (ReSA)
- Oracle Retail Invoice Matching (ReIM)
- Oracle Retail Price Management (RPM)

**Note:** Although Oracle Retail Allocation is a Merchandising application, it is not represented in this batch schedule because it does not have any batch programs to run. All Allocation processing is online processing.

Program List

The columns of the program list provide details about each batch program, as follows:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program name</td>
<td>Name of the program or script</td>
</tr>
<tr>
<td>Functional area</td>
<td>Functional area of the application for which the batch program is run</td>
</tr>
<tr>
<td>Threaded</td>
<td>Whether the program is threaded (Y/N)</td>
</tr>
<tr>
<td>Driver</td>
<td>Program driver</td>
</tr>
<tr>
<td>Phase</td>
<td>Phase during which the program is run (see the batch schedule diagram)</td>
</tr>
<tr>
<td>Pre-dependency</td>
<td>Programs that must be completed before the program can be run</td>
</tr>
<tr>
<td>Post-dependency</td>
<td>Programs that must be run after the program completes successfully</td>
</tr>
<tr>
<td>Timing</td>
<td>How often the program is run (for example, daily, weekly, monthly, ad hoc)</td>
</tr>
<tr>
<td>Restart/Recovery</td>
<td>Whether the program uses restart/recovery (R=Yes, N=No)</td>
</tr>
<tr>
<td>Run Parameters for Program</td>
<td>Command syntax to run the program</td>
</tr>
</tbody>
</table>
For example, the following shows the information in the program list about an RMS phase 3 program named dealday:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Program Name</td>
<td>dealday</td>
</tr>
<tr>
<td>Functional Area</td>
<td>Deals</td>
</tr>
<tr>
<td>Threaded</td>
<td>Y</td>
</tr>
<tr>
<td>Driver</td>
<td>Location</td>
</tr>
<tr>
<td>Phase</td>
<td>3</td>
</tr>
<tr>
<td>Pre-dependency</td>
<td>dealinc, dealfinc, prepost dealday pre</td>
</tr>
<tr>
<td>Post-dependency</td>
<td>prepost dealday post, salmnth</td>
</tr>
<tr>
<td>Timing</td>
<td>Monthly</td>
</tr>
<tr>
<td>Restart/Recovery</td>
<td>R</td>
</tr>
<tr>
<td>Usage</td>
<td>dealday userid/passwd</td>
</tr>
</tbody>
</table>

The program list is grouped in the following order:

- RMS, RTM, and ReSA programs
- RPM programs
- ReIM programs
- RMS extracts for Retail Predictive Application Server (RPAS)
- RMS extracts for Retail Data Warehouse (RDW)

The extracts for RPAS and RDW are programs that are part of the RMS application.
Batch Schedule Diagram

The batch schedule diagram illustrates the program list pre- and post-dependency details. The layout and notations of the diagram also illustrate required sequences and other processing details. Executing the Merchandising batch processing in the manner diagrammed ensures that all critical dependencies are met.

For ease of setting up a schedule at client site, and also based on logical application dependencies, the diagram is divided into three main sections:

- RMS, RTM, ReIM
- ReSA
- RPM

Later chapters of this document show data flow diagrams for other batch processes:

- Chapter 4 shows the Retail Extract, Transform, and Load (RETL) data flows for the extracts from RMS to RPAS.
- Chapter 5 shows the RETL dimension and fact data flows for the extracts from RMS to Oracle Retail Data Warehouse (RDW).
- Chapter 6 shows the RETL data flow for the Promotion dimension extract from RPM to RDW.
- Chapter 7 shows the RETL data flow for the Supplier Invoice Cost dimension extract from ReIM to RDW.
- Chapter 8 shows the RETL data flows for the extracts from RMS to Oracle Retail Advanced Inventory Planning (AIP).

RMS, ReIM, RTM Section

The first section diagrams the RMS, ReIM, and RTM programs and their dependencies. This section is further divided into phases 0 through 7, ad hoc, and date set batch.

Each phase must be completed before the next phase can begin. Also, a phase may contain programs that depend on other programs within the phase. Programs within each phase may need to run in a particular sequence.

The following are brief descriptions of the Merchandising batch processing phases. Depending on your implementation, some programs and phases may not apply.

<table>
<thead>
<tr>
<th>Phase</th>
<th>Description</th>
</tr>
</thead>
</table>
| Phase 0 | The first phase performs essential table maintenance including:  
- Daily purges  
- Updates to currency exchange rates  
- Updates to value-added tax (VAT) data |
| Phase 1 | This phase prepares the tables for interfacing with external systems in Phase 2. Among other programs, the stock variance (stkvar) batch program is run to update stock counts. |
| Phase 2 | During this phase, information is uploaded from external interfaces, including point of sale (POS) data (posupld batch program). |
| Phase 3 | In this phase, the main RMS processing programs are run for purchasing, ordering, stock ledger, deals, and replenishment. |
### Phase Description

<table>
<thead>
<tr>
<th>Phase</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 4</td>
<td>This phase pushes data to external sources. Changed system information is rebuilt. Open to buy (OTB) data is updated. Information is sent to the forecasting system.</td>
</tr>
<tr>
<td>Phase 5</td>
<td>This phase consists of ReIM process upload programs.</td>
</tr>
<tr>
<td>Phase 6</td>
<td>This phase consists of ReIM process roll-up programs.</td>
</tr>
<tr>
<td>Phase 7</td>
<td>This phase consists of ReIM process download programs.</td>
</tr>
<tr>
<td>Ad Hoc</td>
<td>Ad hoc batch programs can be run at any time. The ad hoc programs have no phase dependencies.</td>
</tr>
</tbody>
</table>
| Date Set | The Date Set phase increments the system date and updates other calendar dates.  
**Note:** The date set phase should be the very last phase to run. Even the ad hoc programs should be run before the date set program. |

Read the batch schedule diagram from left to right. In the following example, any of the programs (ediupavl, edupack, stkvar, ditinsrt, lifstkup, ReceiverAdjustment, DiscrepancyPurge) can start at the same time; however, the stkupld program cannot start until the lifstkup program is successfully completed.

#### Sequence

```
Sequence ----------►

<table>
<thead>
<tr>
<th>Phase 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ediupavl</td>
</tr>
<tr>
<td>edupack</td>
</tr>
<tr>
<td>stkvar</td>
</tr>
<tr>
<td>ditinsrt</td>
</tr>
<tr>
<td>lifstkup</td>
</tr>
<tr>
<td>ReceiverAdjustment (ReIM)</td>
</tr>
<tr>
<td>DiscrepancyPurge(ReIM)</td>
</tr>
</tbody>
</table>
```

### ReSA Section

This section diagrams the ReSA programs and their dependencies.

### RPM Section

This section diagrams the RPM programs and their dependencies.
Notations in the Batch Schedule Diagram

Pipes

Pipes are vertical bars (|) that represent the dependencies within a phase. Reading left to right, a pipe indicates that one or more programs to the right depend upon completion of one or more programs to the left.

In the following example, the stkupld module depends on the lifstkup module; that is, the stkupld module can be run only after successful completion of the lifstkup module.

| lifstkup | stkupld |

In the following example, both of the modules cntrordb and reqext are dependent on ociroq. Neither cntrordb nor reqext can be run until the ociroq module has completed successfully.

| ociroq | cntrordb | reqext |

In the following example, the ibcalc module is dependent on both ibexpl and cntrprss. The ibcalc module cannot be run until both ibexpl and cntrprss have completed successfully.

| ibexpl | cntrprss | ibcalc |

Abbreviations

In the diagram, abbreviations in parentheses that follow program names have the following meanings:

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>(perl)</td>
<td>The module is a Perl script.</td>
</tr>
<tr>
<td>(FIF)</td>
<td>The module is related to the Financials application.</td>
</tr>
<tr>
<td>(sqlldr)</td>
<td>There is a sqlloader process to load/ftp the output files.</td>
</tr>
<tr>
<td>(rebuild all)</td>
<td>There is a rebuild process inside the application.</td>
</tr>
<tr>
<td>(IM)</td>
<td>The module is related to Invoice Matching but owned by RMS.</td>
</tr>
<tr>
<td>(RMS)</td>
<td>The module belongs to RMS.</td>
</tr>
<tr>
<td>(RMS)</td>
<td>(Bold type) The RMS module is executed externally to that phase.</td>
</tr>
<tr>
<td>(ReSA)</td>
<td>The module belongs to ReSA.</td>
</tr>
<tr>
<td>(ReSA)</td>
<td>(Bold type) The ReSA module is executed externally to that phase.</td>
</tr>
<tr>
<td>(RelM)</td>
<td>The module belongs to RelM.</td>
</tr>
<tr>
<td>(RTM)</td>
<td>The module belongs to RTM.</td>
</tr>
<tr>
<td>(Weekly)</td>
<td>The module is executed weekly.</td>
</tr>
<tr>
<td>(Monthly)</td>
<td>The module is executed monthly.</td>
</tr>
<tr>
<td>(Forms Auditing)</td>
<td>This is an online forms auditing process related to ReSA.</td>
</tr>
</tbody>
</table>

Footnotes

Footnote symbols (*, **, †, ‡) refer to footnotes that appear below that phase or section of the diagram.
prepost Program

The prepost program 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 have been processed. The prepost program must be run before, after, or both before and after, programs that require specific processing to run or complete successfully.

In the batch schedule diagram, the prepost program is indicated by “pre” and “post” entries, as in the following examples.

In the following example, preprocessing is required before running the ociroq program.

| pre | ociroq |

In the following example, preprocessing is required before running the stkupd program. Also, post-processing is required after successful completion of the stkupd program.

| pre | stkupd | post |

In the following example, post-processing is required after successful completion of the scext program.

| scext | post |
Modifications to the Batch Schedule

The integrated Merchandising batch schedule shows the dependencies for all the programs that could be run by a retailer. Based on many factors, there will always be some programs that a retailer does not run. Determining which programs, or groups of programs, are not required is a job that should be performed at implementation time.

One major factor involves the applications that the retailer has purchased and wants to install:

- For example, a retailer may have purchased RMS, but not ReIM; in this case, the ReIM programs would not be run.
- Another example is that a retailer may not want to use some functionality within an application. Perhaps a retailer purchased RMS but did not purchase the RDW application. In this case, the retailer may not want to run the programs that extract RMS data to be used later by the RDW application.

These major configuration choices also affect whether some programs are used:

- Whether the Retail Integration Bus (RIB) is used
  - For more information about configuring the RIB for Merchandising applications, see “Configuring RPM without the RIB” in the “Backend System Administration and Configuration” chapter of the Oracle Retail Price Management Operations Guide.
- Whether full-featured or simplified Retail Price Management (RPM) is used
  - For more information about configuring simplified RPM, see the “Backend System Administration and Configuration” chapter in the Oracle Retail Price Management Operations Guide.
- Whether full-featured or simplified RTM is used
  - For more information about configuring simplified RTM, see the “Oracle Retail Trade Management Batch” chapter in Volume 1 of the Oracle Retail Merchandising System Operations Guide.
<table>
<thead>
<tr>
<th>Program Name</th>
<th>Functional Area</th>
<th>Threaded</th>
<th>Driver</th>
<th>Phase</th>
<th>Pre-dependency</th>
<th>Post-dependency</th>
<th>Timing</th>
<th>Use Restart/Frequency</th>
<th>Rsc Parameters</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ccprg</td>
<td>Costing</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>daily</td>
<td>N</td>
<td>card</td>
<td></td>
</tr>
<tr>
<td>dfrtbld</td>
<td>Item Maintenance</td>
<td>Y</td>
<td>Dept</td>
<td>3</td>
<td>N</td>
<td>NA</td>
<td>daily</td>
<td>N</td>
<td>card</td>
<td></td>
</tr>
<tr>
<td>discotbapply</td>
<td>Pricing/Transfers/Allocation</td>
<td>Y</td>
<td>Store</td>
<td>3</td>
<td>N/A</td>
<td>NA</td>
<td>daily</td>
<td>R</td>
<td>card</td>
<td></td>
</tr>
</tbody>
</table>
| dealact      | Deals           | Y        | Deal Id| 3     | N/A           | NA             | daily  | N                    | dealact userid/passwd 
|              |                 |          |       |       |                |                 |       |                      |               |       |
| dealinc      | Deals           | Y        | Deal Id| 3     | N/A           | NA             | daily  | N                    | dealinc userid/passwd 
|              |                 |          |       |       |                |                 |       |                      |               |       |
| dealprg      | Deals           | N        | N/A    | NA    | NA            | NA             | monthly| R                    | dealprg userid/passwd 
<p>| | | | | | | | | | | |
|              |                 |          |       |       |                |                 |       |                      |               |       |</p>
<table>
<thead>
<tr>
<th>Program</th>
<th>Trade Management</th>
<th>Y</th>
<th>File-based</th>
<th>ad hoc</th>
<th>N/A</th>
<th>ad hoc</th>
<th>R</th>
<th>This will user/passwd/permission_instruction_file in place of file country_conf_file_params, pre or post processing parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>posupld</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>suppladd</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>rplext</td>
<td>Replenishment</td>
<td>Y</td>
<td>Dept</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>repladj</td>
<td>Replenishment</td>
<td>Y</td>
<td>Dept</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>replsplit</td>
<td>Replenishment</td>
<td>Y</td>
<td>Location</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>rplapprv</td>
<td>Replenishment</td>
<td>N</td>
<td>Dept</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>rplprg</td>
<td>Replenishment</td>
<td>N</td>
<td>Dept</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>rpmmovavg</td>
<td>Pricing</td>
<td>Y</td>
<td>Store/Day</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>rplbhst</td>
<td>Replenishment</td>
<td>N</td>
<td>Dept</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>rtvprg</td>
<td>RTV</td>
<td>N</td>
<td>Dept</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>saexpach</td>
<td>Sales Audit</td>
<td>N/A</td>
<td>SA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>saexpim</td>
<td>Sales Audit</td>
<td>N/A</td>
<td>SA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>saexprms</td>
<td>Sales Audit</td>
<td>Y</td>
<td>Store/Day</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sacrypt</td>
<td>Sales Audit</td>
<td>Y</td>
<td>Store/Day</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sarules</td>
<td>Sales Audit</td>
<td>SA</td>
<td>Store/Day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sarules</td>
<td>Sales Audit</td>
<td>SA</td>
<td>Store/Day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sarules</td>
<td>Sales Audit</td>
<td>SA</td>
<td>Store/Day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sarules</td>
<td>Sales Audit</td>
<td>SA</td>
<td>Store/Day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sarules</td>
<td>Sales Audit</td>
<td>SA</td>
<td>Store/Day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sarules</td>
<td>Sales Audit</td>
<td>SA</td>
<td>Store/Day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sarules</td>
<td>Sales Audit</td>
<td>SA</td>
<td>Store/Day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>sarules</td>
<td>Sales Audit</td>
<td>SA</td>
<td>Store/Day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>This program is executed only once per day. This program is a part of Trade Management Y, File-based, pre/post processing parameters.</td>
</tr>
<tr>
<td>Program Name</td>
<td>Functional Area</td>
<td>Threaded Driver</td>
<td>Phase</td>
<td>Pre-dependency</td>
<td>Post-dependency</td>
<td>Timing</td>
<td>Uses Restart/Recovery</td>
<td>Run Parameters for Programs</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------</td>
<td>-----------------</td>
<td>-------</td>
<td>----------------</td>
<td>-----------------</td>
<td>-------</td>
<td>----------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>ItemReclassBatch</td>
<td>Future Retail</td>
<td>N/A</td>
<td>N/A</td>
<td>reclsdly (RMS)</td>
<td>NewItemLocBatch</td>
<td>daily/ad hoc</td>
<td>rpm-app-userid password</td>
<td></td>
</tr>
<tr>
<td>Program Name</td>
<td>Functional Area</td>
<td>Threaded Driver</td>
<td>Phase</td>
<td>Pre-dependency</td>
<td>Post-dependency</td>
<td>Timing</td>
<td>Use Restart/Recovery</td>
<td>Run Parameters for Programs</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------</td>
<td>-----------------</td>
<td>-------</td>
<td>----------------</td>
<td>-----------------</td>
<td>-------</td>
<td>----------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>cmptrprcildex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>B refer to RDW operations guide</td>
</tr>
<tr>
<td>cstisldex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>C refer to RDW operations guide</td>
</tr>
<tr>
<td>ivaildex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, salstage (RMS), mrt (RMS)</td>
<td>N</td>
<td>N</td>
<td>C refer to RDW operations guide</td>
</tr>
<tr>
<td>lptotcldex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, saexprdw (ReSA), resa2rdw</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>rplapprv (RMS)</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>rplapprv (RMS)</td>
<td>Y</td>
<td>N</td>
<td>Y refer to RDW operations guide</td>
</tr>
<tr>
<td>ivaildex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, salstage (RMS)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>scrqtldex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, salstage (RMS)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>scrtlldex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, rplapprv (RMS), cntrprss (RMS), rplbld</td>
<td>Y</td>
<td>N</td>
<td>Y refer to RDW operations guide</td>
</tr>
<tr>
<td>slsildmex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, saexprdw (ReSA), resa2rdw</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>slsmkdnildex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, salstage (RMS)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>stlblmthex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, salmth (RMS)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>stlblwex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, salweek (RMS)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>vchreschdex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>B, savouch (ReSA)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>vchrmoveldsgex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>B, savouch (ReSA)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>wfslsildex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, salstage (RMS)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
<tr>
<td>wfslsmkdnildex.ksh</td>
<td>RDW interface</td>
<td>N</td>
<td>NA</td>
<td>NA</td>
<td>C, salstage (RMS)</td>
<td>N</td>
<td>N</td>
<td>N refer to RDW operations guide</td>
</tr>
</tbody>
</table>

Notes:
- A list of batch processes on the RDW system.
- The last 10 of the following RDW batch modules:
  - moceckns
  - ralceckns
  - rsmflagns
  - rsmflagltns
  - rsmflagltp
  - rsmflaghlt
  - rsmflagnt
  - rsmflag
  - rsmflag
  - rsmflag
  - rsmflag

**System:**
- Development:
  - System Name: RDW
  - Code Name: RDW
  - Version: 1.0
  - Date: 2023-01-01

**Description:**
- The RDW system is a set of batch processes on the RDW system.
- The last 10 of the following RDW batch modules:
  - moceckns
  - ralceckns
  - rsmflagns
  - rsmflagltns
  - rsmflagltp
  - rsmflaghlt
  - rsmflagnt
  - rsmflag
  - rsmflag
  - rsmflag

**Parameters:**
- A list of batch processes on the RDW system.
- The last 10 of the following RDW batch modules:
  - moceckns
  - ralceckns
  - rsmflagns
  - rsmflagltns
  - rsmflagltp
  - rsmflaghlt
  - rsmflagnt
  - rsmflag
  - rsmflag
  - rsmflag

**Environment:**
- Development:
  - System Name: RDW
  - Code Name: RDW
  - Version: 1.0
  - Date: 2023-01-01
### RMS to AIP RETL Extracts Dependency and Scheduling Details (EXTRACTS FOR AIP)

<table>
<thead>
<tr>
<th>Program Name</th>
<th>Functional Area</th>
<th>Threaded</th>
<th>Driver</th>
<th>Phase</th>
<th>Pre-dependency</th>
<th>Post-dependency</th>
<th>Timing</th>
<th>Uses Restart/Recovery</th>
<th>Run Parameters for Programs</th>
</tr>
</thead>
<tbody>
<tr>
<td>pre_rmse_aip.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_alloc_in_well.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_cl_po.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_future_delivery_alloc.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_future_delivery_order.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, vrplbld, cntrordb</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_item_loc_traits.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, dlyprg</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_item_master.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, reclsdly</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_item_retail.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, dlyprg</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_item_sale.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, sitmain</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_item_supp_country.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, dlyprg</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_merchhier.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, dlyprg</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_orghier.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, dlyprg</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_rec_qty.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, vrplbld, cntrordb, reqext</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_store.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, storeadd, likestore, dlyprg</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_substitute_items.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_suppliers.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_tsf_in_well.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, reqext</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_aip_wh.ksh</td>
<td>AIP Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>pre_rmse_aip.ksh, whadd and dlyprg</td>
<td>Refers to AIP Operations and Installation Guides</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
</tbody>
</table>

### Allocation Program Dependency and Scheduling Details

<table>
<thead>
<tr>
<th>Program Name</th>
<th>Functional Area</th>
<th>Threaded</th>
<th>Driver</th>
<th>Phase</th>
<th>Pre-dependency</th>
<th>Post-dependency</th>
<th>Timing</th>
<th>Uses Restart/Recovery</th>
<th>Run Parameters for Programs</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllocSchedulerBatch.ksh</td>
<td>Scheduled Allocation</td>
<td>Y</td>
<td>Item</td>
<td>Y</td>
<td>N/A</td>
<td>N/A</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
</tbody>
</table>

### RMS to MFP RETL Extracts Dependency and Scheduling Details

<table>
<thead>
<tr>
<th>Program Name</th>
<th>Functional Area</th>
<th>Threaded</th>
<th>Driver</th>
<th>Phase</th>
<th>Pre-dependency</th>
<th>Post-dependency</th>
<th>Timing</th>
<th>Uses Restart/Recovery</th>
<th>Run Parameters for Programs</th>
</tr>
</thead>
<tbody>
<tr>
<td>pre_rmse_rpas.ksh</td>
<td>Planning/Forecast System Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>ftmdnld.ksh</td>
<td>Planning/Forecast System Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_rpas_merchhier.ksh</td>
<td>Planning/Forecast System Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_rpas_orghier.ksh</td>
<td>Planning/Forecast System Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_rpas_store.ksh</td>
<td>Planning/Forecast System Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
<tr>
<td>rmse_rpas_wh.ksh</td>
<td>Planning/Forecast System Interface</td>
<td>N</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Daily</td>
<td>N</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Note:
I - 'I'ntial load
W - 'W'eekly load
F - 'F'ull load
D - 'D'elta load
P - 'P'artial load
I - 'I'ntial load
W - 'W'eekly load

F - multi-threaded full extract if ITEM_LOC is partitioned; single-threaded full extract if ITEM_LOC is not partitioned
D - single-threaded delta extract

Integrated Merchandising Batch Schedule

Phase 1

Phase 2

Date Set

Phase 4

Phase 0

Phase 3

Note:
The prepost pre batch cycle should be run before the batch cycle starts to turn off security.
The prepost post batch cycle should be run after the entire batch cycle is finished to turn security back on.

Ad Hoc Batch

costeventprg posgpdld reimpurge(ReIM) ordinvupld posupld post discotbapply hstbld_diff salins stkvar ibexpl pre ibcalc cntrmain ediupadd pre rplatupd post rilmaint post repladj cntrordb post rpl_pre rplext post pre supsplit cntrprss pre edidlprd post hstbldmth_dif

lifstkup stkupld auditsys

otbdnld ccprg

hstwkupd

reimdiscrepancyupld (ReIM)

batch_compeffupd pre batch_ordcostcompu

batch_alloctsfupd.ksh hstmthupd (sqlldr) edidladd

batch_depchrgupd.ksh lcadnld lcmt700 (perl) edidlcon

batch_expprofupd.ksh wasteadj stkxpld pre stkupd post lcmdnld lcmt707(perl) ediprg

batch_itmcostcompupd.ksh pre fcstrprg post dfrtbld (sqlldr) batch_orpos_extract.ksh (shell)*** post poscdnld post ftmednld fcthreadexec fcexec PriceEventExecutionBatch(RPM) distropcpub txrposdn tifposdn post genpreiss fcstrbld post fcstrbld_sbc hstprg

pre onordext onictext onorddnld hstprg_diff invclshp (IM) * soutdnld cmpupld stlgdnld invaprg

cednld(RTM) nwpyearend storeadd post likestore post salstage saldly stkdly pre vendinvc* post pre salweek ** post salmth post lcmt730(perl) lcupld rpmmovavg *** If RPM pricing info is reqd then run after extraction script 'RPMtoORPOSPublishExport.sh' lclrbld sitmain lcmt798(perl) lcup798 fifgldn1(FIF) pre vendinvf* post salapnd fifgldn3(FIF) *** This script should run before batch 'posdnld' is executed.
mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

dealcls post

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.

rtvprg salprgschedprg

dealinc pre dealfct pre dealday post

mrtprg mrt mrtrtv mrtupd

vrplbld post ordprg invprg(IM)

** Execute prepost with parameters batch_costcompupd post pre † dealact dealfinc supcnstr rplsplit pre rplapprv

supmth post ordautcl otbprg otbupfwd otbupld posdnld postrplprg

** salweek is also dependent on dealfct and dealinc.
Because RMS is the retailer’s central merchandising transactional processing system, it is the principle source of the foundation data needed in some of the Oracle Retail suite of products. RMS provides foundation data to RPAS, and RPAS provides planning data to RMS.

This chapter presents flow diagrams for data processing from sources. The source system’s program or output file is illustrated, along with the program or process that interfaces with the source. After initial interface processing of the source, the diagrams illustrate the flow of the data.

Before setting up a program schedule, familiarize yourself with the functional and technical constraints associated with each program. Refer to the Oracle Retail Merchandising System Operations Guide for more information about these interface programs.
RMS Pre/Post Extract Diagrams

*Note: The pre_rmse_rpasp.ksh program checks for existing .txt output files. Because of this validation, retailers running the program for the first time should include an optional -c parameter. This parameter allows the program to run successfully without pre-existing .txt output files.*
RMS Foundation Data Extract Diagrams

Merchandise Hierarchy for RPAS

RMS EXT 1

sitmain (RMS)

recsdlv (RMS)

dlvpgr (RMS)

rmse_rpas_merchhier.ksh

rmse_rpas_item_master.ksh

rmse_rpas_domain.ksh

rmse_rpas_supplier.ksh

rmse_rpas_attributes.ksh

rmse_rpas_merchhier.dat

rmse_rpas_item_master.dat

rmse_rpas_domain.dat

rmse_rpas_supplier.dat

rmse_rpas_attributes.dat

TO RPAS

TO RPAS

TO RPAS

TO RPAS

TO RPAS

TO RPAS

*Note: The rmse_rpas_attributes.ksh flow is applicable only if issues are active.
**RMS Fact Data Extract Diagrams**

**Sales Extracts For RPAS**

*Note:
If issues are active, the following two files result from the
rmse_rpas_stock_on_hand_ksh flow:
rmse_rpas_stock_on_hand_issues.dat
rmse_rpas_stock_on_hand_sales.dat

If issues are not active, the following file results from the
rmse_rpas_stock_on_hand_ksh flow:
rmse_rpas_stock_on_hand_sales.dat

**Note:
Depending upon the configuration of rmse_rpas_daily_sales.ksh,
the data can be pulled from TRAN_DATA_HISTORY or TRAN_DATA.*
RPAS-RMS Fact Load Diagram

*Note:
? can represent the following:
- i (for issues)
- s (for stores)

?? represents domain 01-99.
Interface Diagrams for RMS and MFP

Because RMS is the retailer’s central merchandising transactional processing system, it is the principle source of the foundation data needed in some of the Oracle Retail suite of products. RMS provides foundation data to RPAS, and RPAS provides planning data to RMS.

This chapter presents flow diagrams for data processing from sources. The source system’s program or output file is illustrated, along with the program or process that interfaces with the source. After initial interface processing of the source, the diagrams illustrate the flow of the data.

Before setting up a program schedule, familiarize yourself with the functional and technical constraints associated with each program. Refer to the Oracle Retail Merchandising System Operations Guide for more information about these interface programs.
**RMS Pre/Post Extract Diagrams**

**RMS Pre RETL Extract Maintenance**

*Note: The pre_rmse_rpas.ksh program checks for existing .txt output files. Because of this validation, retailers running the program for the first time should include an optional -c parameter. This parameter allows the program to run successfully without pre-existing .txt output files.*
RMS Foundation Data Extract Diagrams

Merchandise Hierarchy for MFP

- RMS EXT
- sitmaia (RMS)
- recledty (RMS)
- dyprg (RMS)

\[ \text{rmse_rpas_merchise.ksh} \]
\[ \text{rmse_rpas_item_.master.ksh} \]
\[ \text{rmse_rpas_attributes_ksh*} \]
\[ \text{rmse_rpas_merchise.dat} \]
\[ \text{rmse_rpas_item_.master.dat} \]
\[ \text{rmse_rpas_attributes_dat} \]

TO MFP

*Note: The rmse_rpas_attributes.ksh flow is applicable only if issues are active.
Organization Hierarchy for MFP

storeadd (RMS) -> RMS EXT 1
  
  dlyprg (RMS) -> RMS EXT 1
  
  dlyprg (RMS) -> recstaty (RMS)

Store extracts rmse_rpap_storeksh

rmse_rpap_store.dat

TO RPAS

Organization hierarchy rmse_rpap_orghierksh

rmse_rpap_orghier.dat

TO RPAS

warehouse extracts rmse_rpap_wfhksh

rmse_rpap_wfh.dat

TO RPAS

Time Extract

RMS EXT 1

calendar ftmedelqpc

rmse_rpap_cindmsitfdat

TO RPAS
RMS Fact Data Extract Diagrams

Integration Extracts for MFP

Note:
I is for initial load and W is for weekly load.
Interface Diagrams for RMS and RDW

RMS works in conjunction with the Oracle Retail Extract Transform and Load (RETL) framework. RETL provides high-performance processing to extract data from Oracle Retail applications for use in data warehouses. The architecture allows database batch processes to take advantage of parallel processing capabilities.

This chapter presents flow diagrams for the RETL extraction RMS programs. The source system’s program or output file is illustrated, along with the program or process that interfaces with the source. Note that the data flows are organized according to the logic (dimension data and table data) of Oracle Retail Data Warehouse (RDW), but you can use the data to suit your business needs.

For detailed information about dimensions and facts, see the Oracle Retail Data Warehouse Operations Guide.

For summary information about the configuration, architecture, and features of RETL programs utilized in RMS/ReSA extractions, see the Oracle Retail Management System Operations Guide Volume 3—Backend Configuration and Operations. For more information about the RETL tool, see the current RETL Programmer’s Guide.

Legend

- A: Signifies the completion of RDW maintenance jobs
- B: Signifies the completion of pre-batch maintenance jobs that set system variables
- C: Signifies the completion of pre-batch maintenance jobs that create temporary tables
- RMS or ReSA module on which RETL extract modules are dependent
**RDW Maintenance**

- factopendm.ksh
- medfactopendm.ksh
- factclosedm.ksh
- mt_prime.ksh

**Pre-Batch Maintenance**

1. A
2. pre_dwi_extract.ksh
3. salmth.pc (RMB)
4. pre_dwi_temp.ksh
5. B
6. C

**Post-Batch Maintenance**

- All batch extracts
- post_dwi_temp.ksh

**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. If the retailer does not use RDW, these jobs are not required.

**Note:**

- salmth.pc resets the last eom_date. Thus, it must be run after the system indicator is extracted by pre_dwi_extract.ksh.
Fact Dataflows

**Cost**
- C
  - cstisldex.ksh (DWI)
    - scerx.txt (RMS)
      - cstisddm.txt

**Competitor Pricing**
- B
  - cmptrprcildx.ksh (DWI)
    - cmptprcilddm.txt

**RPM Pricing**
- C
  - prcildex.ksh (DWI)
    - prcilddm.txt

**Stock Ledger**
- C
  - salweek.pc (RMS)
    - stlblwex.ksh (DWI)
      - stlblwdm.txt
  - stlblmth.ksh (DWI)
    - stlblmthdm.txt

**Note:**
Run stock ledger fact loads once weekly.
Fact Dataflows

Supplier Availability
- cntrprss.pc (RMS)
- edupaxl.pc (RMS)
- rplapprv.pc (RMS)
- snavlde.ksh (DWI)

Supplier Contract
- rplapprv.pc (RMS)
- cntrprss.pc (RMS)
- rplbd.pc (RMS)
- rplprg.pc (RMS)
- cntrmain.pc (RMS)
- hsuplld.pc (RMS)

Return to Vendor
- ivridex.ksh (DVI)
- ivriddm.txt

Net Cost
- costcalc.pc (RMS)
- ncstulde.ksh (DVI)
- ncstulddm.txt
Fact Dataflows

**Exchange Rates**

1. B
2. exchngratex.ksh (DWI)
3. exchngratedm.txt

**Sales Forecasts**

1. B
2. fcalupld.pc (RMS)
3. sfciwex.ksh (DWI)
4. sfciwdm.txt

**Note:**
Run sales forecast fact loads once weekly.
Fact Dataflows

Supplier Compliance

Delivery Timeliness

- scrlldex.ksh (DWH)
- scrllddm.txt

Delivery Quantities

- scrtldex.ksh (DWH)
- scrtlddm.txt

Missed Shipments

- scmioldex.ksh (DWH)
- scmiolddm.txt

Missed Purchase Orders

- scmilddex.ksh (DWH)
- scmildddm.txt
Interface Diagram for RPM and RDW

This following program flow diagram shows the RETL extraction program that extracts the Promotion dimension from RPM through the Data Warehouse Interface (DWI). The diagram shows the output files and the scripts that interface with the source. Note that the outputs are based on the logic (dimension data and table data) of Oracle Retail Data Warehouse (RDW), but you can use the data to suit your business needs.

For detailed information about dimensions and facts, see the Oracle Retail Data Warehouse Operations Guide.

See the Oracle Retail Merchandising System Operations Guide Volume 1—Batch Overviews and Designs for more information about the modules shown in the following diagram.

Legend

A  Signifies the completion of pre-batch maintenance jobs

B  Signifies the completion of pre-DWI maintenance module to set system variables

Program Flow Diagram

Promotion Dimension

A  B

prmdtlex.ksh (DWI)  prmdtldm.txt

prmhdrex.ksh (DWI)  prmdtldm.txt

prmrdtm.txt  prmdtldm.txt

prmrdtm.txt
Interface Diagram for RPM and RDW

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. If the retailer does not use RDW, these jobs are not required.

RDW Maintenance

- factspendmksh
- medfactspendmksh
- factclosedmksh
- mt_prmzmksh

Pre-Batch Maintenance

- pre_dwi_extractksh
- pre_dwi_extractksh

Post-Batch Maintenance

- All batch extracts
- post_dwi_tempksh

Note: salminpc resets the last stmr_date. Thus, it must be run after the system indicator is extracted by pre_dwi_extractksh.
Interface Diagram for ReIM and RDW

This following program flow diagram shows the RETL extraction program that extracts the Promotion dimension from ReIM through the Data Warehouse Interface (DWI). The diagram shows the output files and the scripts that interface with the source. Note that the outputs are based on the logic (dimension data and table data) of Oracle Retail Data Warehouse (RDW), but you can use the data to suit your business needs.

For detailed information about dimensions and facts, see the Oracle Retail Data Warehouse Operations Guide.

See the Oracle Retail Merchandising System Operations Guide Volume 1—Batch Overviews and Designs for more information about the modules shown in the following diagram.

Legend

- RMS module on which RETL extract modules are dependent
- Signifies the completion of pre-DWI maintenance module to create currency conversion tables

Program Flow Diagram

Supplier Invoice Cost
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. If the retailer does not use RDW, these jobs are not required.

---

Note: If a job resets the last run date, then it must be run after the system indicator is extracted by pre_dw_extract_ksh.

---

Oracle Retail Merchandising
Interface Diagrams for RMS and AIP

This chapter presents flow diagrams for RETL extract data processing from RMS to AIP. The RMS program or output file is illustrated, along with the program or process that interfaces with the source. The diagrams illustrate the flow of the data after initial interface processing of the source.

Before setting up a program schedule, familiarize yourself with the functional and technical constraints associated with each program. See the Oracle Retail Merchandising System Operations Guide Volume 1—Batch Overviews and Designs for more information about the modules shown in the following diagrams.
RMS Pre/Post Extract Diagrams

RMS Pre RETL Extract Maintenance

Batch Start
(RMS) → pre_rmse_aip.ksh*

next_vdate.txt
last_except
_closed_post_date.txt
last_except
_received_post_date.txt
class
_level_vat
_in.txt
domain
_level.txt
last_eom
date.txt
max_backpost
day.txt
multi_currency
_in.txt
vat_in.txt
prime
(exchange)
rate.txt
prime
_currency
code.txt
skidgr
_vat inc
_rate_in.txt
vdate.txt
curr_bom
data.txt
consolidation
code.txt

max 00:00:00 1
RMS Foundation Data Extract Diagrams

IP = Time-phased inventory planning tool
IP = Time-phased inventory planning tool
IP = Time-phased inventory planning tool
IP = Time-phased inventory planning tool
IP = Time-phased inventory planning tool