# Retek Merchandising

## Release Notes

**Retek Merchandising System**
**Retek Sales Audit**
**Retek Invoice Matching**
**Retek Trade Management**

# Retek Merchandising System (RMS)

## RMS Functional Enhancements

Many new enhancements and features have been added to Retek Merchandising System 9.0. Following the current trend toward Internet computing, this release of RMS is web architected. We have also made navigation easier by adopting a folder and forms based system. New font sizes and a new color palette also make the application more user-friendly.

There have been functional enhancements in the following areas:

- Item maintenance
- Style/color management
- Pricing
- Competitive pricing
- Purchasing
- Deals management
- Inventory management
- Ticketing
- Security
- Sales tax
- Point of sale
- Location lists

### *Item Maintenance*

Many of the new features in the item maintenance area of RMS 9.0 are tailored for the grocery industry. The additions are in the following areas:

- Default unit of purchase
- Decimal size
- Variable weight UPC and EAN
- Wastage attributes

### *Default Unit of Purchase*

The default unit of purchase feature allows you to sell items at one unit of measure, but order items in a different unit of measure. For example, you may sell some kinds of fruit by the pound, but you may order the fruit by the case. When you create an item, you specify the default unit of purchase. The default unit of purchase can be the standard unit of measure for the item, case, or pallet.

When you order the item, the item is ordered at the default unit of purchase and not the standard unit of measure.

### *Decimal Size*

In RMS 9.0, you can now use decimal numbers for a case size. For example, you receive a case of chicken breasts weighing 13.5 pounds. The case has 45 chicken breasts weighing 0.3 pounds each. You can set the standard unit of measure for the chicken breasts to pound and then sell them by the pound, using decimals to enter the weight. The decimal size functionality is available in the ordering and receiving areas of RMS 9.0.

### *Variable Weight UPC and EAN*

In RMS 9.0, you can create and download variable weight UPCs and EANs to the point of sale and scales that communicate information about the weight of purchase for the product. With this feature, you can maintain UPCs and EANs for items and the amount paid for the item at the point of sale based on the purchased weight. You specify the origin of the UPC, either the buyer or the supplier. A bar code is generated for the item at the scale that can be scanned at the register.

### *Wastage Attributes*

In RMS 9.0, you can record wastage attributes for grocery items. You then can designate whether the wastage is due to spoilage or sales. You can also enter a wastage percentage for the item.

#### Sales Wastage

Sales wastage applies to items that shrink during processing. For sales wastage, the actual sales quantity is increased by the wastage percentage so that inventory is decreased by the correct amount. This ensures that replenishment of the item will be based on the sales of the preprocessed item and not on the processed item where some wastage has occurred.

#### Spoilage Wastage

Spoilage wastage applies to items that naturally lose mass as they age. For spoilage wastage, the stock quantity for the item is reduced by the wastage percentage everyday to reflect the average daily loss of inventory. You can also maintain a default daily wastage percent for a spoilage wastage item. This percentage is used when you create a new item/location relationships for the item if you do not specify a different daily wastage percentage.

## *Style/Color Management*

The style/color management feature in RMS 9.0 allows you to work with a specific color of a style in the following functional areas:

- Transfer allocations
- Promotions
- Clearance
- Pricing
- Purchase orders
- Contracts
- Seasons and phases of a fashion style
- Transfers

With this feature, you no longer need to create different SKUs for different colors of the same style.

## *Pricing*

RMS 9.0 has been enhanced to support:

- What-if price change analysis
- Clearance reset date
- Store lead time for price zone changes

### *What-If Price Change Analysis*

You can analyze the impact of a price change before implementing it. You can look at the effects of the proposed change through forecasted sales figures and planned receipts. You can also see the total impact to the price and to the stock on hand using an online worksheet. RMS 9.0 also calculates the retail impact of a promotion based on the estimated sales. You can view this information in RMS 9.0 to create a promotion with the most profitable sale.

### *Clearance Reset Date*

You no longer have to specify a clearance reset date when you create a clearance item. Items can remain on clearance indefinitely so that they can be phased out of inventory permanently.

### *Store Lead Time for Price Zone Changes*

You can now specify when a price zone change takes effect for a store. This allows time for the store personnel to re-ticket items in accordance with the price changes that may occur.

## *Competitive Pricing*

The Competitive Pricing module allows you to track pricing at competitors' stores. You can define the competitors, competitor stores, and pricing information at the SKU/competitor/competitor store/day level. This information allows you to make more informed decisions when pricing your own products.

You also can create competitive shopping lists and historical records of competitors' prices. Each competitor's store must be associated with a specific competitor organization and can also be associated as a competitor of one or more of your company's stores. You can create competitive shopping lists, guides for obtaining competitor pricing information, from a standard RMS item list. Information about the individuals who perform your competitive shopping can also be stored.

The typical workflow for Competitive Pricing is described below.

1. Set up competitor records on RMS.
2. Set up competitor store records on RMS.
3. Associate competitor stores with your merchandising stores.
4. Set up competitive shopper records on RMS.
5. Create item lists to use as shopping lists.
6. Create shopping lists from item lists.
7. Enter competitor prices obtained from competitive shopping.
8. Review competitor price histories to assist in determining your pricing policies.
9. Edit pricing information that was incorrectly entered on RMS.

## *Purchasing*

### *Direct Store Delivery*

The direct store delivery feature allows you to create a purchase order, ship the order, and receive the order all at once. This feature is useful when a buyer has already purchased and received the goods from a supplier, such as a potato chip vendor. You can also indicate when the vendor invoice was paid using the invoice matching module.

### *Quick Order Entry*

You can use the quick order entry feature for direct-to-store orders that only need to be approved and received. Shipments are not created for quick order entries.

### *Redistribution of Approved Orders*

You can now redistribute approved orders without going through the approval process again. You can select the items on an order that must be redistributed and the locations. Deals are automatically reapplied to the order when it is distributed. The original OTB calculations are updated at a later time for the redistributed items.

### *Temporary Distribution of Approved Orders*

You can choose to distribute an order to a redistribution, or virtual, warehouse. A redistribution warehouse does not actually hold stock and cannot be used as a distribution center. It does allow you to create a purchase order before the distribution center is known. The order must be redistributed to actual warehouses or stores when the locations become known and before the merchandise is shipped.

## *Deals Management*

Deals management replaces the deals, brackets, and allowances functionality from previous releases of RMS. Deals management allows you to create and maintain deals with deal partners and fixed deals with suppliers.

### *Fixed Deals with Suppliers*

With fixed deals, your organization receives payments from suppliers in return for mentioning their products in promotions or for displaying their products on prime shelf space. You can create fixed deals and set up a collection schedule from which you can make claims against the supplier and receive payments. Fixed deal payments can be claimed from the supplier either periodically or all at once.

Fixed deals are typically created in Active status; however, you can create a fixed deal in Inactive status. Inactive status holds the claim in merchandising. The Fixed Deal Report provides you with information about active claims for the week.

### *Deals with Deal Partners*

You can create deals with deal partners for items or items at specific locations. Deal partners can be suppliers, wholesalers, distributors, and manufacturers. You can also create PO-specific deals that are only applied to the items on a single purchase order. Within a deal, you create deal components, specify the items for the deal component, and define thresholds.

Deal components are the discounts or rebates that are included in the deal. When you enter deal component information, you determine the billing type, the collection dates, how the deal component is calculated, and the value of the discount or rebate from the deal partner. You can create multiple deal components for a single deal.

After you create the deal components, you select the items to which the deal components are applied. You can also specify specific locations for the items. Items and items at specific locations can also be excluded from deal components.

When you define thresholds, you specify how much must be purchased or sold in order to receive a discount or rebate. You also indicate the value of the discount or rebate.

When you work with purchase orders, you can view all of the deal components that apply to an individual item on the purchase order. You can also view the projected cost of the item with all the valid deal components applied.

## *Inventory Management*

The following inventory management functionality in RMS 9.0 has been updated with the following features:

- Enhanced supplier minimum order requirements
- Due order processing
- Supplier constraint scaling
- Enhanced replenishment orders
- Single location orders
- Warehouse/store assignments
- Warehouse demand replenishment
- Receiving
- Product transformation

### *Enhanced Supplier Minimum Order Requirements*

You can define minimum order requirements for purchase orders for the supplier. You cannot approve purchase orders that do not meet the minimum order requirements of the supplier. You can specify up to two minimum order requirements and indicate whether a purchase order has to meet one or both of the requirements. You also can specify that purchase orders that do not meet the minimum order requirements are purged.

### *Due Order Processing*

Due order processing allows replenishment purchase orders to be created based on the need for all of the items from a supplier, as opposed to the need of an item at a single location. Often, due order processing results in more efficient ordering because you can obtain cost discounts by meeting supplier minimums while maintaining a desired service level, build orders to meet multiple truckloads, and meet supplier minimums to reduce freight charges. You can indicate whether due order processing is performed for items from a supplier or items in a department from a supplier, and then the automatic replenishment process determines when an order is due.

You can select the level at which the due order processing is based. When due order processing is performed at the purchase order level, the decision to order is based on the combined service level of all the items on the order. An item is considered due when the accepted stock out level of the item is less than the estimated stock out level for the item, if the item is not ordered today. At a purchase order level, the stock out and expected stock out levels for all the items on the order are summed and compared to determine if the purchase order as a whole is due. When due order processing is performed at the item level, a replenishment purchase order is due when the stock out level is less than the estimated stock out level for any one item on the purchase order.

### *Supplier Constraint Scaling*

Scaling is available for purchase orders created manually and automatically through the replenishment process. With scaling, purchaser orders are increased or decreased within the scaling constraints in order to meet purchasing objectives and increase the cost effectiveness of the ordering process. You can indicate that purchase orders for the supplier are scaled and the objective of the scaling process. You can select whether the objective of scaling is to get as close as possible to the maximums or minimums defined for the scaling constraints.

When you use order scaling, you define two scaling constraints. You select the type of constraint and the minimum and maximum values for the constraint. Depending on the objective of the scaling process, purchase orders are scaled as close a possible to the maximum or minimum value of the scaling constraint. You can also enter the allowable deviation for the purchase order from the value of the scaling constraint.

### *Enhanced Replenishment Orders*

You can indicate whether replenishment orders should be automatically generated in approved status or automatically generated but requiring approval. When orders are created through the replenishment process, you can review the orders, change parameters such as vendor constraints and item/location replenishment parameters, and recalculate the orders based on the changes you have made.

You can also define how pack size rounding is performed. You can round to the case, pallet, or both case and pallet. You can also create replenishment orders for single locations. This is helpful when ordering in truckloads where the supplier only delivers to one location.

You can maintain the inventory management information at either the supplier or supplier/department level. This information is used for replenishment orders; however, you can edit this information at the order level.

### *Single Location Orders*

You can indicate that replenishment orders must be created for single locations. This helps in building truckloads where a truck, due to distances involved, will only deliver to one location.

In RMS 8.0, replenishment orders included as many locations as were valid as long as the supplier and department (for departmental orders) were the same. The locations could be a combination of crossdock warehouses, warehouses (for warehouse stocked merchandise) and stores.

In RMS 9.0, you can indicate that the orders for a specific supplier must be single location orders. This functionality is available for automatic replenishment only.

### *Warehouse/Store Assignments*

You can assign one or multiple stores to a warehouse. You also indicate the date on which the assignment begins. RMS verifies that assignment dates are not in conflict with any existing assignments. If there is a conflict, you can view the conflicting assignments.

### *Warehouse Demand Replenishment*

When you use Retek Demand Forecasting with RMS 9.0, you can create replenishment forecasts for warehouses. You can use the following two forecasting methods:

- Time supply
- Dynamic

These methods still rely on store stock positions to produce accurate order quantities for the warehouse.

### *Receiving*

You can receive items that were not entered on the original purchase order. This might happen when you receive a substitute item or when a pack item is received with the wrong components. Substitute items are supported in batch receiving. To receive pack items with incorrect components, you can receive the component items that were delivered to you instead of the pack item.

### *Product Transformation*

You can transform one product into several products, and apportion the costs of the original product across the resulting products at various rates. You decide the calculation for the cost distribution. These transactions are normal stock adjustments in the stock ledger. This feature allows you to purchase a side of beef and then transform it into multiple products for sale.

## *Ticketing*

The ticketing feature in RMS 9.0 has been expanded to better support grocery retailers and organizations that operate in multiple countries. These additions include:

- Grocery ticket attributes
- Shelf edge labels
- Multiple sizes on tickets
- Multiple currencies on tickets

### *Grocery Ticket Attributes*

Ticket attributes are used to calculate the price per unit of measure. The price per unit of measure can then be printed on tickets. You can enter the unit of measure (UOM), the size of the item in the specified unit of measure (Size of Each), and the size that the item will be priced at on the ticket in the unit of measure selected (Size of Price Per). The price by unit of measure is calculated as follows:

Price / (Size of Each / Size of Price Per)

For example, if you have a five pound bag of rice that you are selling for $3.20US, you would enter the following information so that the price per ounce is displayed on the ticket:

- The unit of measure is ounces.
- The size of each bag of rice in ounces is 80, as there are 80 ounces in five pounds.
- The size that the item will be priced at is 1 because you want the price of the item per one ounce printed on the ticket.

In addition to the other ticket information, the ticket will display that the price of the bag of rice is $0.04 per ounce.

### *Shelf Edge Labels*

You can specify that a ticket is a shelf edge label. This feature accommodates many types of shelf labels, allowing you to create labels for many different types of displays and fixtures.

### *Multiple Sizes on Tickets*

You can print multiple sizes on a single ticket. You can define which sizes and the number of different sizes to be printed on the ticket. This feature allows retailers to customize tickets for international business.

### *Multiple Currencies on Tickets*

You can specify a secondary currency for a store. Then, when tickets are printed for an item, the price of the item is printed in both the primary and secondary currency of the store.

## *Security*

The RMS 9.0 security features provides a method for protecting product, location, and price zone records. A database administrator sets up the security information for groups. Users with limited access are linked to one or more groups. The users within the groups share the same permissions in the specified functional areas at the merchandise level, organizational level, and price zone level.

### *Product Security*

The product security feature limits user access to functional areas by department, class, subclass, style, and SKU. You can limit access to the following functional areas:

- Pricing
- Costing
- Promotion
- Clearance
- Transfers
- Allocations
- Orders
- Stock Counts

### *Location Security*

The location security feature limits user access to functional areas by region, district, store, and warehouse. You can limit access to the following functional areas:

- Promotions
- Transfers (to and from)
- Allocations (to and from)
- Shipments
- Orders
- Stock Counts
- Ticket Requests
- Inventory Adjustments
- Returns to Vendors (RTV)

### *Price Zone Security*

The price zone security feature limits user access to functional areas by price zone group and price zone level. You can limit access to the following functional areas:

- Pricing
- Clearance

## *Sales Tax*

The sales tax feature allows you to specify the tax codes that apply to an item at a specific store. To ensure that stores have accurate sales tax information, the tax codes are sent through the nightly batch process two days in advance of when they are effective at the stores.

When you define sales tax information, you must first add geocodes to RMS. A geocode is a geographical region, such as a state or county, with its own tax codes. You can add geocodes for the following:

- Countries
- States
- Counties
- Cities
- Districts

You then create the tax codes. A tax code consists of a tax jurisdiction and a tax type with associated tax rates. The tax code determines the tax rate that is applied to an item at the store. You can then associate tax codes and stores with a geocode and associate tax codes with items.

The association of the tax codes and stores with a geocode determines which tax codes are effective at a specific store. You associate stores with a geocode. You then associate the tax codes with a geocode. As a result, all of the tax codes associated with a geocode are effective at all of the stores associated with a geocode. When an item is sold at a specific store, the tax rates of the applicable tax codes are applied to the item.

You apply a tax code to items, departments, and item lists. You can create tax product group item lists that include items that share the same tax codes. The tax codes that are ultimately applied to an item at a store are determined by the tax codes that the item shares with the store. The tax codes of a store are determined by the tax codes that are associated with the geocode to which the store belongs.

You can view the tax rates that apply to the item at a specific store, or you can view all of the tax codes associated with an item, department, or item list. You can only view item lists that are tax product groups.

You can also generate four types of sales tax reports:

- Tax Rate Report - This report displays the tax jurisdictions, tax levels, tax types, and tax rates that apply to the item, department, or item list.
- Items With No Tax Codes - This report displays all of the items without a tax code applied to them at a store or geocode level. You can generate the report by store or by geocode.
- Tax Rates by Location - This report displays the geocode level, stores, and the tax codes effective at the stores. You can generate the report by store or by geocode.
- Tax Rates by Item/Department - This report displays the geocode level, stores, departments or items, and the tax codes associated with the item or department.

### *Point of Sale Configurations*

With the point of sale (POS) feature in RMS 9.0, you can create store configurations that are downloaded to the store through a standard POS download process. The configurations are:

- Coupons - You can create and maintain configurations for expense coupons. You can link coupons to items through the merchandise hierarchy.
- POS Buttons - You can create and maintain touch button configurations for the POS register. Each touch button configuration can be associated with a single item or set up as a programmable button.
- Product Restrictions - You can create and maintain restrictions for products, such as minimum age requirements, tender type restrictions, and date and time restrictions. You can link product restriction to items through the merchandise hierarchy.
- Tender Type - You can create and maintain acceptable tender types for each store.
- Paid In/Paid Out - You can create and maintain acceptable paid-in/paid-out transaction types for each store.
- Money Orders - You can create and maintain a money order configuration for stores that sell money orders. The information in the configuration includes tax class, maximum face value of the money order, and associated fees.
- Supplier Payment Types - You can create and maintain valid methods by which a supplier can be paid. This information is sent to the store so that store employees can pay the supplier directly with the correct payment type.

### *Location Lists*

The location list feature allows you to group stores, warehouses, or combination of stores and warehouses. The list is identified by one ID. You can add locations to a list one by one, or you can enter criteria by which the locations should be grouped. Location lists make it easy to apply changes to multiple locations at one time.

Location lists can be used when you:

- Edit warehouse and store attributes by item or by item list.
- Identify store locations for a fashion, pack, or staple item.
- Add stores to a promotion.
- Add stores to a price change event.
- Apply a mask to an order, allocation, transfer, or contract.
- Add stores and warehouses to a mask.
- Edit replenishment parameters by item or item list.
- View replenishment parameters by item.
- Request a stock count for warehouses and stores.
- Edit the shipping schedule from a warehouse to multiple stores.
- Edit the retail price of a store-priced item.
- Add stores to a store grade in a store grade group.

You can use a location list to make mass changes to:

- Price and cost zone groups
- Store attributes

## Location traits

# RMS Technical Enhancements

## *Oracle 8i*

RMS has been upgraded to run on Oracle 8*i*. Oracle 8*i* is designed for efficient application development, superior content management, and successful application deployment on the Internet and therefore will allow Retek to further exploit the unlimited opportunities the Internet provides. Oracle 8*i* provides the advanced tools to manage all types of data that is found in today's most popular web sites and also the performance and scalability needed to support these large sites and other mission critical applications.

Oracle 8i is also introducing the Oracle Enterprise Manager, which is the comprehensive management framework for managing the Oracle database and application environment. Oracle Enterprise Manager includes an easy to-use centralized console, a rich set of management tools, and the extensibility to detect and solve problems that arise. It also includes several administrative applications for performing day-to day tasks for databases and applications, such as scheduling and backup routines.

## *Forms*

Retek has also upgraded to Oracle Developer 6.0. This version of Oracle Developer leverages the Internet platform to provide business solutions that automatically scale and perform with a minimum development effort.

Oracle developer provides the following benefits:

- Leverage the power of a rapid application development (RAD) environment that allows for an easy transition from a client/server environment to a three-tiered Internet environment.
- Deliver scalable enterprise-class applications around the world automatically allowing companies to meet the growing demands of their business, choose the most optimal deployment architecture, and deliver multilingual applications around the globe.
- Deliver solutions in web time to easily build complex database applications that automatically scale, realize benefits of optimized deployment platform tightly integrated with a modeling and declarative suite, promote standards-based, team development with reusable components, and build rich user interfaces.
- Protect your investment to provide support for emerging technologies and extend your pre-packaged applications
- Achieve open access to support multiple platforms and multiple databases
- Leverage the Internet platform to exploit the power of Oracle 8*i*.

## RMS 9.0 Operations Guide Changes

The notes in this section describe major modifications made to the operations guide for RMS 9.0 and the core merchandising transaction systems including:

- Retek Sales Audit
- Retek Invoice Matching
- Retek Trade Management

### *New Dataflow Overviews*

The guide now contains overviews of RMS interface dataflows:

- Posting Sales Data to RMS
- RMS-electronic data interchange (EDI)
- RMS-Invoice Matching
- RMS-Oracle Financial Application 10.7 (general ledger, accounts payable)
- RMS-RDF
- RMS-RPP
- RMS-RDM
- RMS-RDW Interface
- RMS-Retek Trade Management (RTM)
- Retek Sales Audit (ReSA)
- Updating the point-of-sale system (POS) from RMS

### *RMS Deals*

dealcalc.pc—Calculates the net cost, net net cost, and dead net net cost for all items that are on the deal_sku_temp table
dealcls.pc—Closes deals that have reached their end dates.
dealprg.pc—Purges deals after they have been held in the system for the specified number of history months after they are closed.
ditinsrt.pc—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.
orddscnt.pc—Finds discounts that apply to purchase orders and inserts the discounts for the items on the order into the ordsku_discount table

### *Location and Product Level Security*

slocrbld.pc—Maintains location security data.
sprdrbld.pc—Maintains product security data.
szonrbld.pc—Maintains zone security data.

### *Replenishment*

supcstrr.pc—supplier restraint scaling
supcnstr.pc—processs all orders eligible for scaling during the nightly replenishment run

### *Pricing*

pctrandn.pc—price transactions download for store zone changes; writes price changes for a particular store that has been rezoned.
pctranex.pc—performs the final steps in processing pricing transactions that are created when a store is moved from one zone to another.

### *Point of Sale and Tax*

poscdnld.doc—download of all Retek POS configurations' from the RMS database to a flat file for loading into a POS/Back Office application
grpdnld.doc—download program to download group number/department information to individual Point-of-Sale (POS) systems
gcdupld.doc—upload geocodes from an outside source into the RMS tables
txrtuupld.doc—upload tax rates from an outside source into the tables into RMS. The upload provides the means to create new tax jurisdictions, tax codes, and tax rates. In addition it inserts new tax codes and rates and updates the rates for existing tax codes
tifposdn.doc—processes records off the TIF_EXPLODE table, calculates a tax rate total for every item/site combination, and writes the total to a POS flat file
txrposdn.doc—processes records from the geocode_txcde, product_taxcode, and tax_rates tables, and writes all item/site combinations to the TIF_EXPLODE table for tax rate processing to be sent to the POS flat file

### *Support for Oracle Financial Application version 10.7*

The operations guide reflects modifications made in support of Oracle Financial Application version 10.7, specifically general ledger and accounts payable. These modified and new batch programs provide integration with Oracle Financial Application 10.7:

- fifcuru2107.pc—Processes currency exchange rates from Oracle to RMS
- fifvndu2107.pc—Processes suppliers and supplier addresses from Oracle to RMS
- fiftrmu2107.pc—Processes freight and payment terms from Oracle to RMS
- prepost.pc (with the argument `post-fifgldn107`)— Processes shipment receipts to Oracle from RMS
  Used when Invoice Matching is unavailable
- fifinvcu107.pc—Processes matched invoice data from Retek Invoice Matching (RMS) staging tables to Oracle staging tables
- fifgldn1107.pc—Processes daily sales transaction data at the SKU/location level from RMS to Oracle
- fifgldn2107.pc—Processes daily sales transaction data rolled up to the department/class/subclass level for location from RMS to Oracle

- fifgldn3107.pc—Processes monthly sales transaction data rolled up to department/class/subclass level for location from RMS to Oracle
- saexpgl107.pc—ReSA totals export to general ledger

### *Retek Sales Audit Batch Modules*

Retek Sales Audit (ReSA) has been almost completely redeveloped from its previous version. Design documents are included in this guide for the batch modules listed here:

saexpach.pc—Export to ACH
saexprdw.pc—Export to RDW
saexprms.pc—Export to RMS
saexpuar.pc—Export to Driscoll UAR
saexpsfm.pc—Transfers store day fuel and fuel tank transactions to the Oracle Site Fuels Management application
samastersfm.pc—Sends reference (stores, items) data from ReSA to Oracle Site Fuels Management
sagetref.pc—Fetches all reference information needed by saimplog.pc and writes this information out to separate output files
saimpadj.pc—Processes adjustments from applications. Creates new records for all export programs that transfer adjusted data, with the exception of the application that provided the adjustment
saimptlog.pc—Imports and processes RTLOGs. Validates transactions and writes errors for reconciliation through the interactive (online) audit 9.0 supports DSD transactions for vend #, vend inv #, pay ref #, proof del. #
saimptlogfin.pc—Creates balances (over or under) by store, register, or cashier. Marks the store day record in the ReSA import log as partially or fully loaded. Unlocks store day records after all store transactions are imported
sapreexp.pc—Tracks changes in totals previously exported. Writes changes to revision tables recognized by the export programs for re-export to applications
sapurge.pc—Controls the size of the sales audit tables by purging, or deleting, older information
sarules.pc—Processes new or edited rules created and used in the interactive audit
sastdycr.pc—Creates store/day, import log and export log records
satotals.pc—Processes new or edited totals created and used in the interactive audit
savouch.pc—Processes voucher transactions (e.g., gift certificates)
saescheat.pc—Sends escheatment totals for each state or country as defined by the retailer to Retek Invoice Matching
saexpgl107.pc—Exports totals to Oracle Financial Application version 10.7 General Ledger for sales transactions, tenders, paid-ins, and paid-outs
saexpim.pc—Exports to Retek Invoice Matching: invoice number, vendor number, payment reference number, proof of delivery number, and payment date

### *Trade Management Batch Modules*

RTM 9.0 supports customs entry, harmonized tariff schedules, letter of credit, and transport transactions through the following batch programs and Perl scripts:

- htsupld.pc—Processes a file containing the most recent United States Customs tariff schedule to RMS tables
- cednld.pc—Customs entry download to brokers
- lcadndld.pc—Downloads approved letter of credit applications to a bank in a Retek format
- lcmdnld.pc—Downloads amended letter of credit information to a bank
- lcmt707 Perl script
- lcmt730 Perl script—Converts letter of credit confirmations from a S.W.I.F.T. (MT730) format to a Retek format suitable for uploading into RMS
- lcmt798 Perl script—Converts letter of credit drawdowns and bank fee data from a S.W.I.F.T. (MT798) format to a Retek format suitable for uploading into RMS
- lcupld.pc—Writes data from the file converted by lcmt730.perl to the RMS table lc_head
- lcup798.pc—Writes data from the file converted by lcmt798.perl to the RMS table `LC_ACTIVITY`
- lcmt700 Perl script—Converts approved letter of credit applications from a Retek format to the S.W.I.F.T. (MT700) format
- prepost.pc (with the argument `htsupld_pre()`)—Truncates the table populated by htsupld.pc
- tranupld.pc—Uploads data from trading partners about the transport of merchandise from the manufacturing site through customs clearance

In addition, RTM now interfaces Retek Invoice Matching for obligations and custom duties.

### *Interface with Retek Predictive Planning*

Retek Merchandising System 9.0 (RMS) interfaces Retek Predictive Planning 9.0 (RPP) in support of open-to-buy (OTB) planning. The customer can choose to send RPP stock ledger data at the subclass/location/week level either for the most current week or for a historical period. The following RMS batch programs support this interface:

- forgdnld.pc
- fwhdnld.pc
- fmednlds.pc
- fmednldf.pc
- ftmednld.pc

- onordext.pc
- stlgdnld.pc
- prepost.pc, with the argument `onordext_pre()`
- otbupld.pc

### *Interface with Retek Demand Forecasting*

RMS 9.0 introduces these enhancements to the Retek Demand Forecasting interface:

**Daily Sales Forecasts—**In addition to the existing weekly sales forecasts, RMS 9.0 now sends RDF 9.0 daily sales transactions and warehouse issues and in return receives daily sales forecasts. Also, RMS sends RDF daily stockouts (stock outages at a store). Because certain areas of RMS do not use daily forecasts, daily and weekly forecasts exist in parallel. RMS' replenishment and scaling modules use the daily forecasts whenever they are available. If the daily forecasts do not cover the time period needed by replenishment, it will revert back to its current method of using 1/7 of the weekly forecast.

**Calendar download**—RMS downloads the customer's 4-5-4 calendar to RDF, thereby no longer requiring RDF to maintain its own calendar. The calendar adjusts for fiscal year start dates.

These RMS batch programs (and one Perl script) support the entire RMS interface to RDF:

- forgdnld.pc
- fwhdnld.pc
- fsadnlds.pc
- fsadnldf.pc
- fisdnlds.pc
- fisdnldf.pc
- fdaydnld.pc
- soutdnld.pc
- fmednlds.pc
- fmednldf.pc
- ftmednld.pc
- fcstrprg.pc
- fcslupld.pc
- fcstrbld.pc
- fdayupld Perl script

### *Interface with Retek Distribution Management*

RMS 9.0 provides the following additional support for Retek Distribution Management:

**Tracking standalone allocations**—Standalone allocations allow a customer to move merchandise stored in a warehouse to other locations. RMS now tracks standalone allocations and updates RDM with that data.

**Catch weights**—RMS no longer needs to send RDM weight values in whole numbers only. Because RDM can now accept weight values that include decimals, RMS no longer truncates the decimal, thereby sending the exact value.

**Multiple UPCs for Items**—RMS now sends RDM the primary UPC for an item and as many additional UPCs as may be defined for that item.

## RMS Known Issues

- RDF cannot be called from within RMS. This issue will be addressed in the first patch.
- Selecting too many items within a multiview causes the application to hang. This is an Oracle issue and will be addressed.
- If a form has a modal window and an alert is received from the window, the application will hang. This is an Oracle issue and will be addressed.
- Closing an alert using the X in the top right corner hangs the application. This is an Oracle issue and will be addressed in a future version of forms.
- Netscape 4.7 - Logon dialog box does not appear until Developer Forms Runtime window is manually received. This is an Oracle issue and will be addressed via Oracle Bug 1031986.

### *Location & Product Security*

- **ISSUE:** A "You do not have sufficient security privileges" message is displayed if the limited security user's default store is not set-up in Location security. This prevents security users from viewing information about themselves in the User Attributes form.

  **RESOLUTION**: Set-up the limited security user's default store in Location Security specifying "store" as the functional area, "store" as the organizational level, and the default store number.

- **ISSUE:** If a limited security user does not have security privileges to all stores in a Price Zone and tries to create a new zone based on this price zone, an "Invalid Store" message is displayed.

  **RESOLUTION**: Use Oracle's other security features to disable the Price Zone menu option, preventing limited security users access to that feature.

- **ISSUE:** Even if you do not change security settings, if you edit the comments on the Security Group Maintenance (sgrpattr) form, security will be rebuilt in the overnight batch process.

  **RESOLUTION:** Works as designed.

- **ISSUE:** If the user-id that executes batch programs is not set-up on the sec_user_privs table, unexpected batch errors will occur and batch processing will not complete successfully.

  **RESOLUTION**: Check with your System Administratror to add the user-id that executes batch programs to location & product security with full privileges.

- **ISSUE:** If limited security users view or edit purchase orders that contain items from multiple departments, they may get incorrect information if they have not been set-up correctly in all security groups.

  **RESOLUTION**: Be sure to set-up users privileges in Product Security for all departments that reference items on the purchase order.

# Retek Sales Audit (ReSA)

## ReSA Functional Enhancements

This document lists the descriptions and benefits of the new enhancements and features of Retek Sales Audit (ReSA) 9.0 pre-release.

The updated ReSA has:

- Seamless communication
- A precise audit process
- Lower cost and less time for system implementation
- Lower maintenance costs
- A central point for data collection

### *Interface: Imports and Exports*

The expanded ReSA interface now includes all Retek Enterprise applications. In addition to Merchandising (RMS) and Data Warehouse (RDW), ReSA provides standard data interfaces to:

- Oracle Site Fuels Management (SFM) — ReSA 9.0 can capture transactions from pump controllers, whether or not the controller is integrated with the POS, and even if there is no POS upload.
- Oracle general ledger (GL) systems
- Account Clearinghouse (ACH) file is provided for interfacing with the banking system.
- Universal Account Reconciliation system (J-Driscoll)
- escheqat
- Retek Invoice Matching (RIM)—ReSA 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 Retek Invoice Matching. RIM then uses that data to create an invoice for DSD within RIM. Data exported to RIM by ReSA include invoice number, vendor number, payment reference number, proof of delivery number, payment date, and paid indicator.
  Note that Retek Invoice Matching matches unpaid invoices that are marked as approved and sends them to Oracle Financial Application 10.7's Accounts Payable for payment.
- Escheatment—The laws of individual states and countries require a retailer to return monies for aged, unclaimed gift certificates and vouchers. This is called "escheatment." ReSA writes records for this data to tables that are read into RIM. The data can then be sent as invoices approved for payment to Oracle Financial Application 10.7's accounts payable module.

### *Adjustments*

ReSA 9.0 can process adjustments imported from external systems.

ReSA treats all adjustments as on-line entries, so that they generate exports to other Retek systems.

### *Variable import files*

For retailers who use trickle polling, ReSA 9.0 can accept a variable number of import files that represent the data for one store day.

### *Multiple Threads processing*

Retailers with large sales volumes can split the consolidated TLOG files for multiple threads processing.

### *Lottery support*

ReSA 9.0 assists retailers who sell lottery tickets. A retailer can use it to define automated lottery audit rules. At store close, ReSA can reconcile lottery activity for the day. It is possible to take an end-of-day reading for each lottery game and produce a total for lottery ticket sales. It then formulates summary transactions from yesterday's and today's numbers.

### *Exports*

All adjustment transactions are made with *Full Disclosure*. When a transaction is reversed, ReSA 9.0 processes a complete reversal of the original transaction followed by the corrected transaction. Consistent with the current Retek Merchandising System, adjustments made for a prior accounting period are reflected in the current accounting period.

ReSA 9.0 can move data to other applications to:

- Compute on-hand data for financial systems
- Evaluate inventory
- Analyze market baskets
- Profile customers
- Detect fraud

The retailer can also create custom exports from the ReSA database tables, such as:

- Commission systems for salesperson compensation
- Payroll systems for money order payments
- Expense A/P system for 1099 vendor information
- Expense A/P system - Money order/Cash paid-out information

- Merchandise A/P system – Payments to Direct Store Delivery (DSD) vendors

### *Master Reference Information*

Reference information is enhanced in ReSA 9.0:

- Bank Attributes — Bank attributes and the stores associated with each bank.
- Store Attributes — Store attributes (G/L cost center number), store groups, data expected to be imported and exported, bank attributes, closed dates for the head office and individual store exceptions (e.g. holidays, closed store days).
- User-defined tables enable users to extract data for external systems. Users can define and maintain tables about POS transactions or totals stored for a specific store day, and use them to correct data received from the stores:
- To create store date details. Store-day tables can be populated with data before the RTLOG files are run.
- Enables user selection of balancing level at store, register or cashier.
- The user can define their escheatment rules.

## ReSA Performance Enhancements

- The Unit of Work (UOW) parameter is a new system variable. The user can set this parameter to either send an entire store/day file to RMS or to send transactions one-by-one. If you choose to send all store transactions at once, all data must be error-free.
- Database Purge — The user can define the period of time that ReSA stores data and when it deletes data.
- Mass Transaction Delete — The user can delete incorrectly polled data.
- Mass Modification — The user can correct transaction level data en masse, such as a change an item number (SKU) or UPC.

### *Reports*

ReSA 9.0 can generate these reports:

- Daily Totals Flash Report — Contains user-definable totals by store.
- The Cashier/Register Close Report — Displays over-short totals by cashier or register, consistent with the retailer's balancing level.
- Store Close Report — Displays over-short totals at the store level.
- Audit Trail Report — Contains the list of store days that were changed in ReSA or by external system within a selected date range.

### *Interactive audit:*

ReSA 9.0 enhances the interactive audit process by enabling an auditor to focus on exceptions.

A store employee can review adjustments made by headquarters. Since totals are stored at four levels, the differences between the totals calculated by ReSA, reported by point-of-sale (POS), reported by the store, or reported by headquarters are clearly visible.

An Auditor can view, edit, or create transactions that were last captured by POS individual vendor pay outs.

Unique functions available in the interactive audit process include:

| | |
|---|---|
| Store Summary | Transaction Search |
| Cashier/Register Level Summary | Transaction Detail Maintenance |
| Error List | Missing Transactions |
| Over/Short | Audit Trails on Transactions |
| Totaling Details | Audit Trails on Totals |
| Item Summary | Tender Summary |

Cashier/Register Review — A cashier can review totals and adjust them as needed.

Day Close — Either the store employee or a store manager will close the store. A store closing signifies that all required errors have been worked (fixed or overridden) before exporting the data in ReSA. Upon completion, the store status will be set to "Closed".

### *Customized setup*

#### *Automated Audit Rules*

The Automated Audit Process is driven by user-defined audit rules that are created and maintained by the retailer. The modular approach to this validation allows great flexibility for a retailer to define rules dynamically and fine-tune the system in order to focus on potential problem areas.

The retailer can set up ReSA 9.0 to define the audit rules to validate data accuracy before the system can pass it to other systems and functions.

The audit rules can be linked and applied to a particular store or store group, and to execute at specified steps during the audit process. The system can set up:

- Error codes and error categories
- Field validations
- Sequence number validation.

ReSA will support both standard audit rules delivered with the system as well as custom audit rules specified by the retailer. Audit rules fall into these types:

- Custom table driven audits
- Custom sub-program driven audits

Retailers also can define automated lottery audit rules.

### *Automatic Totals*

ReSA 9.0 can record four numbers for each total:

- ReSA calculated (totals automatically calculated from POS detail transactions)
- POS reported (totals extracted as reported from POS transactions)
- Store reported (totals entered by store personnel)
- Headquarters reported (any adjustments made by HQ staff or systems)

User-defined totals drive the automated totaling process. Retailers can now define and redefine totals categories, and specify which totals to include in the over short calculation. For example:

- 'Sales' could be one totals category. Totals could represent types of sales merchandising, such as money orders, or lottery.
- Another category could be 'Cash paid-outs.' Totals could represent vendor payouts for direct store delivery transactions, services, lottery wins, payroll or theft losses.
- General Ledger Account Maintenance is available within the Automated Totals module as a cross reference to map merchandise hierarchy levels from ReSA to the appropriate general ledger accounts within Oracle. The accounting user can enter the appropriate accounting fields for a specific total and location.

A retailer can associate totals to stores, so that the expected totals for each store can be calculated and reported. Totals can be used for:

- Data entry
- Automatic calculations of other totals
- Validation
- Calculations of over short (O/S)
- Exports to general ledger (G/L) accounts
- Flash sales reports (see ***Reports***)

### *Employee Authorization*

The enhanced ReSA 9.0 can provide different access points and navigation paths for different users: cashiers, store managers, site and headquarters auditors. To determine the type of work the user can accomplish, ReSA 9.0 checks the user's identity, type, and responsibilities.

For example:

A store employee may access the same windows as a head office auditor, but does not have the authority to modify information or the work sequence.

ReSA 9.0 uses the following information to ascertain a user's role and access to windows and operations:

- Employee information (Oracle login ID, Personal ID number, POS ID number)
- Employee/Store assignment
- Auditor Information (Name, Oracle Login ID, phone extension, e-mail)
- Auditor/Store assignment

## *Content and Revision Control*

ReSA 9.0 tracks:

- Which data files it should receive and export
- The status of each store day’s data
- The current number of errors, by category, for a particular store day
- When it receives and exports data files

## Database Change Script Listing

| TABLES | Upgrade Script(s) |
|---|---|
| ADDR | dbc405.sql<br>dbc570_addr.sql<br>dbc654_addr.dbc |
| ALC_HEAD | dbc504.sql<br>dbc669_alc_head.dbc |
| ALC_HEAD_TEMP | dbc601_new_tables.dbc |
| ALLOC_DETAIL | dbc424.sql<br>dbc564_alloc_detail.sql |
| ALLOC_HEADER | dbc535.sql<br>dbc565.sql |
| ALLOC_HEADER_TEMP | dbc669_alloc_header_temp.dbc |
| ALLOC_REV | dbc426.sql |

| | |
|---|---|
| ARI_INTERFACE_TEST | dbc644_ari_interface_test.dbc |

| | |
|---|---|
| BILL_BACK_PERIOD | dbc408.sql |

| | |
|---|---|
| CE_CHARGES | dbc561_ce_charges.sql<br>dbc575_ce_charges.sql<br>dbc668_ce_charges.dbc |

| | |
|---|---|
| CE_HEAD | dbc503.sql<br>dbc604_ce_head.dbc<br>dbc606_ce_head.dbc<br>dbc614_ce_head.dbc |

| | |
|---|---|
| CE_ORD_ITEM | dbc518.sql<br>dbc607_ce_ord_item.dbc<br>dbc661_ce_ord_item.dbc |

| | |
|---|---|
| CLASS_SALES_FORECAST | dbc666_class_sales_forcast.dbc |

| | |
|---|---|
| CLEAR_RESET_CALC | dbc406.sql |

| | |
|---|---|
| COMPANY_CLOSED | dbc550.sql |

| | |
|---|---|
| COMPANY_CLOSED_EXCEP | dbc550.sql |
| COMPETITOR | dbc301.sql |
| COMPHEAD | dbc545.sql<br><br>dbc669_comphead.dbc |
| COMP_LIST_TEMP | dbc301.sql |
| COMP_PRICE_HIST | dbc301.sql |
| COMP_SHOPPER | dbc301.sql |
| COMP_SHOP_LIST | dbc301.sql |
| COMP_STORE | dbc301.sql |
| COMP_STORE_LINK | dbc301.sql<br><br>dbc668_comp_store_link.dbc |
| CONTEXT_HELP | dbc437.sql |

| | |
|---|---|
| DAILY_ITEM_FORECAST_1 | dbc601_new_tables.dbc |

| | |
|---|---|
| DAILY_ITEM_FORECAST_2 | dbc601_new_tables.dbc |

| | |
|---|---|
| DEAL_COMP_TYPE | dbc408.sql |

| | |
|---|---|
| DEAL_DETAIL | dbc408.sql<br>dbc548.sql<br>dbc600_deal_detail.sql |

| | |
|---|---|
| DEAL_HEAD | dbc408.sql |

| | |
|---|---|
| DEAL_ITEMLOC | dbc668_deal_itemloc.dbc |

| | |
|---|---|
| DEPT_SALES_FORECAST | dbc666_dept_sales_forcast.dbc |

| | |
|---|---|
| DESC_LOOK | dbc436.sql<br>dbc671_desc_look_rms_table_del_aidr.dbc<br>dbc671_desc_look_rms_table_del_aur.dbc |

| | |
|---|---|
| DOC | dbc437.sql |

| | |
|---|---|
| DYNAMIC_HIER_CODE | dbc411.sql<br>dbc431.sql |
| EDI_COST_CHG | dbc438.sql |
| EDI_NEW_ITEM | dbc300.sql<br>dbc401.sql<br>dbc438.sql |
| EXP_PROF_HEAD | dbc427.sql |
| FASHPACK_FILTER_TEMP | dbc582_flashback_filter_temp.sql |
| FIF_CLASS | dbc674_fif_class.dbc |
| FIF_DEPS | dbc675_fif_deps.dbc |
| FIF_LINE_TYPE_XREF | dbc670_fif_line_type.dbc |
| FIF_LOCATION | dbc676_fif_location.dbc |

| | |
|---|---|
| FIF_ORA_CROSS_REF | dbc566_fif_ora_cross_ref.sql<br>dbc584_fif_ora_cross_ref.sql<br>dbc672_fif_ora_cross_ref.dbc |
| FIF_SUBCLASS | dbc677_fif_subclass.dbc |
| FIF_SUPS | dbc418.sql<br>dbc563_fif_sups.sql |
| FIXED_DEAL | dbc301.sql |
| FIXED_DEAL_DATES | dbc301.sql<br>dbc408.sql |
| GEOCODE_STORE | dbc577_triggers.sql |
| HTS_OGA | dbc401.sql |
| IIF_DISCOUNT | dbc601_new_tables.dbc |
| IIF_HEAD | dbc437.sql<br>dbc439.sql |

| | |
|---|---|
| | dbc533.sql<br>dbc571_iif_head.sql<br>dbc605_iif_head.dbc<br>dbc636_iif_head.dbc<br>dbc678_iif_head.dbc |
| IIF_NON_MERCH | dbc544.sql |
| INVC_DISCOUNT | dbc601_new_tables.dbc |
| INVC_HEAD | dbc406.sql<br>dbc530.sql<br>dbc605_invc_head.dbc |
| INVC_MATCH_WKSHT | dbc569_invc_match_wksht.sql |
| INVC_NON_MERCH | dbc532.sql |
| INVC_NON_MERCH_TEMP | dbc601_new_tables.dbc |
| ITEM_EXP_HEAD | dbc438.sql |

| | |
|---|---|
| ITEM_HTS_ASSESS | dbc401.sql |

| | |
|---|---|
| ITEM_SEASONS | dbc412.sql |

| | |
|---|---|
| ITEM_SUPPLIER | dbc642_item_sup_trg.dbc<br><br>dbc662_item_supplier.dbc<br><br>dbc671_item_supplier_rms_col_isp_prisupindvpn_aur.dbc |

| | |
|---|---|
| ITEM_SUPP_COUNTRY | dbc300.sql<br><br>dbc401.sql<br><br>dbc408.sql<br><br>dbc436.sql |

| | |
|---|---|
| LANG | dbc412.sql |

| | |
|---|---|
| LC_ACTIVITY | dbc516.sql<br><br>dbc669_lc_activity.dbc |

| | |
|---|---|
| LC_AMENDMENTS | dbc515.sql |

| | |
|---|---|
| LC_DOWNLOAD | dbc679_lc_download.dbc |

| | |
|---|---|
| LC_HEAD | dbc405.sql<br><br>dbc514.sql |

| | |
|---|---|
| LIF_BOL_CONTAINER_ITEM | dbc576_lif_bol_container_item.sql |

| | |
|---|---|
| LIF_ITEM | dbc603_lif_item.dbc<br><br>dbc622_lif_item.dbc |

| | |
|---|---|
| LIF_SA_ALLOC | dbc601_new_tables.dbc<br><br>dbc646_lif_sa_alloc.dbc<br><br>dbc656_lif_sa_alloc.dbc |

| | |
|---|---|
| LOCATION_CLOSED | dbc551.sql |

| | |
|---|---|
| MC_REJECTIONS | dbc406.sql |

| | |
|---|---|
| MOD_ORDER_ITEM_HTS | dbc601_new_tables.dbc<br><br>dbc667_mod_order_items_hts.dbc |

| | |
|---|---|
| MPF_MIN_MAX | dbc579_mpf_min_max.sql |

| | |
|---|---|
| MULTIVIEW_DEFAULT_40 | dbc400.sql<br>dbc411.sql<br>dbc431.sql |
| MULTIVIEW_SAVED_40 | dbc400.sql<br>dbc411.sql |
| NAV_ELEMENT_MODE | dbc441.sql |
| NAV_FOLDER | dbc441.sql |
| NON_MERCH_CODE_COMP | dbc601_new_tables.dbc |
| NON_MERCH_CODE_HEAD | dbc531.sql |
| ON_ORDER_TEMP | dbc601_new_tables.dbc |
| ORDER_TYPES | dbc535.sql |
| ORDFASH_TEMP | dbc405.sql |
| ORDHEAD | dbc436.sql |

| | |
|---|---|
| | dbc628_ordhead.dbc<br>dbc647_ordhead.dbc |
| ORDHEAD_DISCOUNT | dbc601_new_tables.dbc |
| ORDLOC | dbc412.sql<br>dbc422.sql |
| ORDLOC_REV | dbc423.sql |
| ORDLOC_WKSHT | dbc401.sql |
| ORDSKU | dbc420.sql<br>dbc637_ordsku.dbc |
| ORDSKU_DISCOUNT | dbc408.sql |
| ORDSKU_EXP | dbc438.sql |
| ORDSKU_HTS_ASSESS | dbc438.sql |
| ORDSKU_INVC_COST | dbc568_ordsku_invc_cost.sql |

| | |
|---|---|
| ORDSKU_REV | dbc420.sql |
| ORDSKU_TEMP | dbc420.sql |
| ORD_TEMP | dbc401.sql |
| ORD_XDOCK_TEMP | dbc438.sql |
| PACKHEAD | dbc410.sql<br>dbc613_packhead.dbc |
| PACKSTORE | dbc436.sql<br>dbc577_triggers.sql |
| PACKWH | dbc436.sql |
| PACK_TMPL_HEAD | dbc567_pack_tmpl_head.sql |
| PARTNER | dbc408.sql<br>dbc541.sql<br>dbc569_partner.sql |

| | |
|---|---|
| | dbc633_partner.dbc |
| POS_TENDER_TYPE_HEAD | dbc407.sql |
| PRICE_BATCH_TRAN | dbc403.sql |
| PRICE_CHANGE_SKU_IMPACT | dbc558.sql |
| PRICE_CONFLICT | dbc403.sql |
| PRICE_EVENT_SKU | dbc438.sql |
| PRICE_HIST | dbc401.sql |
| PRICE_OVERLAP_LOG | dbc403.sql |
| PRICE_SUSP_DETAIL | dbc403.sql |
| PRICE_SUSP_HEAD | dbc406.sql |
| PROMHEAD | dbc433.sql |

| | |
|---|---|
| PROMSKU | dbc416.sql |
| PROM_MIX_MATCH_BUY | dbc416.sql |
| PROM_MIX_MATCH_GET | dbc416.sql |
| PROM_THRESHOLD_SKU | dbc416.sql |
| RAG_SKUS | dbc410.sql |
| RAG_SKUS_ST | dbc436.sql<br>dbc577_triggers.sql |
| RAG_SKUS_ST_HIST | dbc634_resize_6_tables.dbc |
| RAG_SKUS_WH | dbc436.sql |
| RAG_SKUS_WH_HIST | dbc634_resize_6_tables.dbc |
| RAG_STYLE | dbc437.sql<br>dbc611_rag_style.dbc |

| | |
|---|---|
| RECEIVING | dbc401.sql |

| | |
|---|---|
| REPL_ATTR_UPDATE_ITEM | dbc415.sql |

| | |
|---|---|
| REPL_ITEM_LOC | dbc415.sql<br>dbc578_repl_item_loc.sql |

| | |
|---|---|
| REPL_RESULTS | dbc429.sql<br>dbc554.sql |

| | |
|---|---|
| REPL_RESULTS_TEMP | dbc554.sql |

| | |
|---|---|
| REQ_DOC | dbc437.sql |

| | |
|---|---|
| RESTART_BOOKMARK | dbc402.sql<br>dbc412.sql |

| | |
|---|---|
| RESTART_CONTROL | dbc402.sql |

| | |
|---|---|
| RESTART_PROGRAM_HISTORY | dbc402.sql<br>dbc412.sql<br>dbc440.sql |

| | |
|---|---|
| | dbc529.sql |

| | |
|---|---|
| RESTART_PROGRAM_STATUS | dbc402.sql<br>dbc412.sql<br>dbc440.sql |

| | |
|---|---|
| RTK_ERRORS | dbc442.sql |

| | |
|---|---|
| RTK_ROLE_PRIVS | dbc438.sql |

| | |
|---|---|
| RTV_DETAIL | dbc437.sql |

| | |
|---|---|
| SA_BALANCE_GROUP | dbc409.sql |

| | |
|---|---|
| SA_BANK_STORE | dbc640_sa_bank_store.dbc |

| | |
|---|---|
| SA_COMMENTS | dbc601_new_tables.dbc |

| | |
|---|---|
| SA_CUSTOMER | dbc409.sql |

| | |
|---|---|
| SA_ERROR_CODES | dbc409.sql |

| | |
|---|---|
| SA_ERROR_TEMP | dbc680_sa_error_temp.dbc |

| | |
|---|---|
| SA_ERROR_WKSHT | dbc601_new_tables.dbc |

| | |
|---|---|
| SA_ESCHEAT_OPTIONS | dbc601_new_tables.dbc<br>dbc602_sa_escheat_options.dbc<br>dbc621_sa_escheat_options.dbc<br>dbc660_sa_escheat_options.dbc |

| | |
|---|---|
| SA_ESCHEAT_TOTAL | dbc534.sql<br>dbc660_sa_escheat_total.dbc |

| | |
|---|---|
| SA_ESCHEAT_VOUCHER | dbc534.sql<br>dbc660_sa_escheat_voucher.dbc |

| | |
|---|---|
| SA_EXPORTED | dbc618_sa_exported.dbc<br>dbc623_sa_exported.dbc<br>dbc660_sa_exported.dbc |

| | |
|---|---|
| SA_EXPORTED_REV | dbc618_sa_exported_rev.dbc<br>dbc623_sa_exported_rev.dbc<br>dbc652_sa_exported_rev.dbc |

| | |
|---|---|
| SA_FIF_ORA_CROSS_REF | dbc601_new_tables.dbc<br><br>dbc631_sa_fif_ora_cross_ref.dbc |

| | |
|---|---|
| SA_FLASH_SALES | dbc409.sql |

| | |
|---|---|
| SA_HQ_VALUE | dbc653_sa_hq_value.dbc |

| | |
|---|---|
| SA_MISSING_TRAN | dbc409.sql<br><br>dbc500.sql |

| | |
|---|---|
| SA_PARM | dbc601_new_tables.dbc |

| | |
|---|---|
| SA_PARM_TYPE | dbc601_new_tables.dbc |

| | |
|---|---|
| SA_POS_VALUE | dbc653_sa_pos_value.dbc |

| | |
|---|---|
| SA_POS_VALUE_WKSHT | dbc601_new_tables.dbc<br><br>dbc624_sa_pos_value_wksht.dbc<br><br>dbc653_sa_pos_value_wksht.dbc<br><br>dbc684_sa_pos_value_wksht.dbc |

| | |
|---|---|
| SA_REALM | dbc585_sa_realm.sql |

| | |
|---|---|
| SA_REALM_TYPE | dbc585_sa_realm_type.sql |

| | |
|---|---|
| SA_REFERENCE | dbc409.sql |

| | |
|---|---|
| SA_RULE_ERRORS | dbc615_sa_rule_errors.dbc |

| | |
|---|---|
| SA_RULE_HEAD | dbc573_sa_rule_head.sql<br><br>dbc657_sa_rule_head.dbc |

| | |
|---|---|
| SA_STORE_DAY | dbc409.sql |

| | |
|---|---|
| SA_STORE_EMP | dbc540.sql |

| | |
|---|---|
| SA_STORE_VALUE | dbc653_sa_store_value.dbc |

| | |
|---|---|
| SA_SYSTEM_OPTIONS | dbc409.sql<br><br>dbc501.sql<br><br>dbc539.sql |

| | |
|---|---|
| SA_SYS_VALUE | dbc653_sa_sys_value.dbc |

| | |
|---|---|
| SA_SYS_VALUE_WKSHT | dbc601_new_tables.dbc<br><br>dbc624_sa_sys_value_wksht.dbc<br><br>dbc653_sa_sys_value_wksht.dbc |

| | |
|---|---|
| SA_TOTAL | dbc609_sa_total.dbc<br><br>dbc660_sa_total.dbc |

| | |
|---|---|
| SA_TOTAL_HEAD | dbc572_sa_total_head.sql |

| | |
|---|---|
| SA_TRAN_DISC | dbc409.sql<br>dbc512.sql<br>dbc527.sql<br>dbc635_sa_tran_disc.dbc |

| | |
|---|---|
| SA_TRAN_DISC_REV | dbc409.sql<br>dbc513.sql<br>dbc528.sql<br>dbc635_sa_tran_disc_rev.dbc |

| | |
|---|---|
| SA_TRAN_HEAD | dbc409.sql<br>dbc508.sql<br>dbc519.sql<br>dbc537.sql<br>dbc610_sa_tran_head.dbc<br>dbc616_sa_tran_head.dbc<br>dbc635_sa_tran_head.dbc<br>dbc648_sa_tran_head.dbc |

| | |
|---|---|
| SA_TRAN_HEAD_REV | dbc409.sql<br>dbc509.sql<br>dbc520.sql<br>dbc538.sql<br>dbc610_sa_tran_head_rev.dbc<br>dbc616_sa_tran_head_rev.dbc<br>dbc635_sa_tran_head_rev.dbc |

| | |
|---|---|
| SA_TRAN_ITEM | dbc409.sql<br>dbc510.sql<br>dbc521.sql<br>dbc635_sa_tran_item.dbc |

| | |
|---|---|
| SA_TRAN_ITEM_REV | dbc409.sql<br>dbc511.sql<br>dbc522.sql<br>dbc635_sa_tran_item_rev.dbc |

| | |
|---|---|
| SA_TRAN_TAX | dbc409.sql<br>dbc559.sql<br>dbc635_sa_tran_tax.dbc |

| | |
|---|---|
| SA_TRAN_TAX_REV | dbc409.sql<br>dbc560.sql<br>dbc635_sa_tran_tax_rev.dbc |

| | |
|---|---|
| SA_TRAN_TENDER | dbc409.sql<br>dbc506.sql<br>dbc525.sql<br>dbc635_sa_tran_tender.dbc |

| | |
|---|---|
| SA_TRAN_TENDER_REV | dbc409.sql<br>dbc507.sql<br>dbc526.sql<br>dbc635_sa_tran_tender_rev.dbc |

| | |
|---|---|
| SA_VOUCHER | dbc409.sql<br>dbc534.sql |

| | |
|---|---|
| SA_VOUCHER_OPTIONS | dbc562_sa_voucher_options.sql |

| | |
|---|---|
| SA_VR_HEAD | dbc586_sa_vr_head.sql |

| | |
|---|---|
| SA_VR_REALM | dbc585_sa_vr_realm.sql |

| | |
|---|---|
| SIZE_TMPL_DETAIL | dbc601_new_tables.dbc<br><br>dbc669_size_tmpl_detail.dbc |

| | |
|---|---|
| SIZE_TMPL_HEAD | dbc601_new_tables.dbc<br><br>dbc669_size_tmpl_head.dbc |

| | |
|---|---|
| SKULIST_DETAIL | dbc434.sql |

| | |
|---|---|
| SKULIST_HEAD | dbc434.sql |

| | |
|---|---|
| SOURCE_DLVRY_SCHED | dbc552.sql |

| | |
|---|---|
| SOURCE_DLVRY_SCHED_DAYS | dbc552.sql<br>dbc667_source_dlvry_sched_days.dbc |

| | |
|---|---|
| SOURCE_DLVRY_SCHED_EXC | dbc552.sql |

| | |
|---|---|
| STORE | dbc409.sql<br>dbc433.sql<br>dbc437.sql<br>dbc645_store.dbc |

| | |
|---|---|
| STORE_ADD | dbc409.sql<br>dbc437.sql<br>dbc556.sql |

| | |
|---|---|
| SUBCLASS_SALES_FORECAST | dbc666_subclass_sales_forcast.dbc |

| | |
|---|---|
| SUB_ITEMS_HEAD | dbc438.sql |

| | |
|---|---|
| SUPS | dbc419.sql<br>dbc549.sql |

| | |
|---|---|
| SUPS_MIN_FAIL | dbc417.sql |

| | |
|---|---|
| SUP_IMPORT_ATTR | dbc517.sql |

| | |
|---|---|
| SUP_INV_MGMT | dbc432.sql |

| | |
|---|---|
| SUP_REPL_DAY | dbc432.sql |

| | |
|---|---|
| SYSTEM_OPTIONS | dbc404.sql<br>dbc502.sql<br>dbc536.sql<br>dbc546.sql<br>dbc555.sql<br>dbc627_system_options.dbc<br>dbc655_system_options.dbc<br>dbc658_system_options.dbc |

| | |
|---|---|
| SYSTEM_VARIABLES | dbc410.sql |

| | |
|---|---|
| TARIFF_TREATMENT | dbc438.sql |

| | |
|---|---|
| TEMP_PACK_TMPL | dbc581_temp_pack_tmpl.sql |

| | |
|---|---|
| TICKET_TYPE_DETAIL | dbc435.sql |

| | |
|---|---|
| TIF_EXPLODE | dbc505.sql<br><br>dbc574_tif_explode.sql |

| | |
|---|---|
| TRANSPORTATION | dbc617_transportation.dbc<br><br>dbc643_transportation.dbc |

| | |
|---|---|
| TRANS_DELIVERY | dbc638_trans_delivery.dbc |

| | |
|---|---|
| TRANS_PACKING | dbc673_trans_packing.dbc |

| | |
|---|---|
| TRAN_DATA_HISTORY | dbc547.sql<br><br>dbc634_resize_6_tables.dbc<br><br>dbc646_tran_data_history.dbc |

| | |
|---|---|
| TSFDETAIL | dbc401.sql |

| | |
|---|---|
| UDA | dbc619_uda.dbc |

| | |
|---|---|
| UNIT_OPTIONS | dbc300.sql |

| | |
|---|---|
| UPC_EAN | dbc608_upc_ean.dbc |

| | |
|---|---|
| USER_ATTRIB | dbc410.sql |

| | |
|---|---|
| VEHICLE_ROUND | dbc681_sa_vehicle_round.dbc |

| | |
|---|---|
| WEEK_DATA | dbc634_resize_6_tables.dbc |

| | |
|---|---|
| WH | dbc406.sql<br>dbc437.sql<br>dbc553.sql |

| | |
|---|---|
| WH_STORE_ASSIGN_TEMP | dbc668_wh_store_assign_temp.dbc |

| | |
|---|---|
| WIN_SKUS | dbc410.sql<br>dbc612_winsku.dbc |

| | |
|---|---|
| WIN_STORE | dbc436.sql |

| | |
|---|---|
| WIN_STORE_HIST | dbc634_resize_6_tables.dbc |

| | |
|---|---|
| WIN_WH | dbc436.sql |

| | |
|---|---|
| WIN_WH_HIST | dbc634_resize_6_tables.dbc |

**SEQUENCES**

| | |
|---|---|
| BILL_OF_LADING_SEQUENCE | dbc682_bill_of_lading_sequence.dbc |
| COMP_SEQUENCE | dbc301.sql |
| COMP_SHOP_SEQUENCE | dbc301.sql |
| COMP_ST_SEQUENCE | dbc301.sql |
| FIXED_DEAL_SEQUENCE | dbc301.sql |
| REQ_DOC_SEQUENCE | dbc437.sql |
| SA_BAL_GROUP_NO_SEQUENCE | dbc409.sql |
| SA_COMMENT_SEQ_NO_SEQUENCE | dbc601_new_seq.sql |
| SA_ESCHEAT_OPT_SEQ_NO_SEQUENCE | dbc557.sql |
| SA_ESCHEAT_SEQ_NO_SEQUENCE | dbc557.sql |

| | |
|---|---|
| SA_PARM_ID_SEQUENCE | dbc601_new_seq.sql |

| | |
|---|---|
| SA_PARM_TYPE_ID_SEQUENCE | dbc601_new_seq.sql |

| | |
|---|---|
| SA_REALM_ID_SEQUENCE | dbc601_new_seq.sql |

**VIEWS**

| | |
|---|---|
| DAILY_ITEM_FORECAST | dbc601_new_views.dbc |

| | |
|---|---|
| V_COLOR_LIST | dbc413.sql |

| | |
|---|---|
| V_PSD_SKUS | dbc403.sql |

| | |
|---|---|
| V_SA_BALANCE_GROUP_CASHIER | dbc543.sql |

| | |
|---|---|
| V_SA_ERROR_ALL | dbc664_v_sa_error_all.dnc<br><br>dbc685_v_sa_error_all.dbc |

| | |
|---|---|
| V_SA_GET_TOTAL | dbc625_v_sa_get_total.dbc<br><br>dbc664_v_sa_get_total.dbc |

| | |
|---|---|
| V_SA_GET_TOTAL_WKSHT | dbc632_v_sa_get_total_wksht.dbc<br><br>dbc664_v_sa_get_total_wksht.dbc |

| | |
|---|---|
| V_SA_PARM_INFO | dbc587_v_sa_parm_info.sql |

| | |
|---|---|
| V_SA_REALM_INFO | dbc601_new_views.dbc |

| | |
|---|---|
| V_SA_TOTAL | dbc641_v_sa_total.dbc<br><br>dbc664_v_sa_total.dbc<br><br>dbc665_v_sa_total.dbc<br><br>dbc685_v_sa_total.dbc |

| | |
|---|---|
| V_SA_TOTAL_AUDIT | dbc651_v_sa_total_audit.dbc<br>dbc664_v_sa_total_audit.dbc<br>dbc665_v_sa_total_audit.dbc<br>dbc685_v_sa_total_audit.dbc |

| | |
|---|---|
| V_SA_TOTAL_STATUS | dbc650_v_sa_total_status.dbc |

| | |
|---|---|
| V_SA_TRAN_ITEM_NET | dbc542.sql |

**DROPPED TABLES**

| | |
|---|---|
| B_ERR_XREF | dbc583_b_err_xref.sql |

| | |
|---|---|
| IF_PACK_SALES | dbc659_if_pack_sales.dbc |

| | |
|---|---|
| MULTIVIEW_DEFAULT_41 | dbc580_multiview_41.sql |

| | |
|---|---|
| MULTIVIEW_SAVED_41 | dbc580_multiview_41.sql |

**DROPPED SEQUENCES**

| | |
|---|---|
| ERROR_ID_SEQUENCE | dbc630_rem_ari_sequence.dbc |
| GROUP_ID_SEQUENCE | dbc630_rem_ari_sequence.dbc |
| STATE_ID_SEQUENCE | dbc630_rem_ari_sequence.dbc |

**DROPPED FUNCTIONS**

| | |
|---|---|
| SA_T_GASDRIVEOF | dbc639_drop_functions.dbc |
| SA_T_GASSALES | dbc639_drop_functions.dbc |
| SA_T_PUMPTEST | dbc639_drop_functions.dbc |

**DROPPED PACKAGE**

| | |
|---|---|
| RMS_ARI_INTERFACE | dbc663_rmsair_drop_package.dbc |