

## **Oracle® Retail Merchandising**

Batch Schedule

Release 12.1

October 2008

Copyright © 2008, Oracle. All rights reserved.

Primary Author: Nathan Young

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

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

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

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

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

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

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

## 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 and Oracle Retail Demand Forecasting 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 Store Inventory Management.
- (v) the software component known as **Crystal Enterprise Professional and/or Crystal Reports Professional** licensed by Business Objects Software Limited (“Business Objects”) 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.



---

---

# Contents

|   |            |
|---|------------|
| <b>Preface .....</b>  | <b>vii</b> |
| Audience .....  | vii        |
| Related Documents .....                                       | vii        |
| Customer Support .....  | viii       |
| Conventions .....   | viii       |
| <b>1 Introduction to Merchandising Batch Processing .....</b> | <b>1</b>   |
| Batch Processing .....  | 1          |
| Types of Batch Programs .....                                 | 1          |
| Batch Window .....  | 2          |
| Batch Schedule and Phases .....                               | 2          |
| Integrated 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          |
| <b>2 Program List .....</b>                                   | <b>11</b>  |
| <b>3 Batch Schedule Diagram .....</b>                         | <b>19</b>  |
| <b>4 Interface Diagrams for RMS and RPAS .....</b>            | <b>21</b>  |
| RMS Pre/Post Extract Diagrams .....                           | 22         |
| RMS Foundation Data Extract Diagrams .....                    | 23         |
| RMS Fact Data Extract Diagrams .....                          | 25         |
| RPAS-RMS Fact Load Diagram .....                              | 26         |
| <b>5 Interface Diagrams for RMS and RDW .....</b>             | <b>27</b>  |
| <b>6 Interface Diagram for RPM and RDW .....</b>              | <b>39</b>  |
| <b>7 Interface Diagram for ReIM and RDW .....</b>             | <b>41</b>  |



---

---

# Preface

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 Merchandising Implementation Guide
- Oracle Retail Merchandising System Operations Guide
- Oracle Retail Price Management Operations Guide
- Oracle Retail Invoice Matching Operations Guide
- Oracle Retail Data Warehouse Operations Guide
- Oracle Retail Predictive Application Server documentation
- Oracle Retail Demand Forecasting documentation

## Customer Support

<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](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:

| Column                     | Description  |
|----------------------------|--|
| Program name               | Name of the program or script  |
| Functional area            | Functional area of the application for which the batch program is run      |
| Threaded                   | Whether the program is threaded (Y/N)                                      |
| Driver                     | Program driver   |
| Phase                      | Phase during which the program is run (see the batch schedule diagram)     |
| Pre-dependency             | Programs that must be completed before the program can be run              |
| Post-dependency            | Programs that must be run after the program completes successfully         |
| Timing                     | How often the program is run (for example, daily, weekly, monthly, ad hoc) |
| Restart/Recovery           | Whether the program uses restart/recovery (R=Yes, N=No)                    |
| Run Parameters for Program | Command syntax to run the program  |

For example, the following shows the information in the program list about an RMS phase 3 program named dealday:

|                  |  |
|------------------|--|
| Program Name     | dealday                                |
| Functional Area  | Deals                                  |
| Threaded         | Y                                      |
| Driver           | Location                               |
| Phase            | 3                                      |
| Pre-dependency   | dealinc, dealfinc, prepost dealday pre |
| Post-dependency  | prepost dealday post, salmnth          |
| Timing           | Monthly                                |
| Restart/Recovery | R                                      |
| Usage            | dealday userid/passwd                  |

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 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.

### 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.

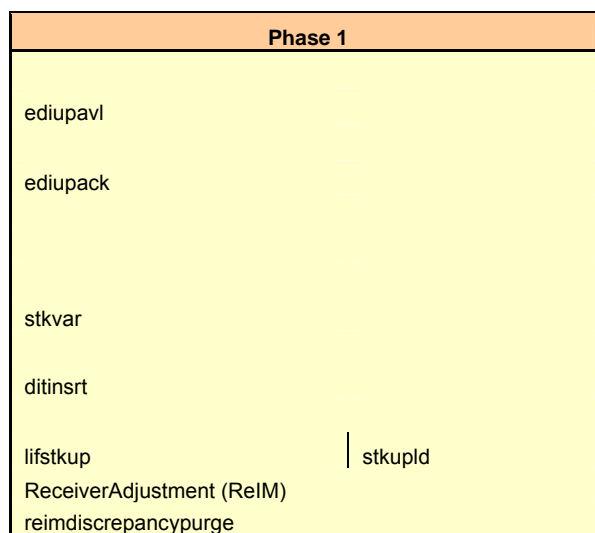
The following are brief descriptions of the Merchandising batch processing phases. Depending on your implementation, some programs and phases may not apply.

| Phase   | Description   |
|---------|---|
| Phase 0 | The first phase performs essential table maintenance including: <ul style="list-style-type: none"> <li>▪ Daily purges</li> <li>▪ Updates to currency exchange rates</li> <li>▪ Updates to value-added tax (VAT) data</li> </ul> |
| 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 4 | 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.  |

| Phase    | Description  |
|----------|--|
| Phase 5  | This phase consists of ReIM process upload programs.   |
| Phase 6  | This phase consists of ReIM process roll-up programs.  |
| Phase 7  | This phase consists of ReIM process download programs.   |
| Ad Hoc   | Ad hoc batch programs can be run at any time. The ad hoc programs have no phase dependencies.  |
| Date Set | The Date Set phase increments the system date and updates other calendar dates.<br><b>Note:</b> 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, ediupack, 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 -----▶



## 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 cntrrordb and reqext are dependent on ociroq. Neither cntrrordb nor reqext can be run until the ociroq module has completed successfully.

|        |                     |
|--------|---------------------|
| ociroq | cntrrordb<br>reqext |
|--------|---------------------|

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

|                   |        |
|-------------------|--------|
| ibexpl<br>cntprss | ibcalc |
|-------------------|--------|

### Abbreviations

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

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

### 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, pre-processing is required before running the ociroq program.

|            |               |
|------------|---------------|
| <b>pre</b> | <b>ociroq</b> |
|------------|---------------|

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

|            |               |             |
|------------|---------------|-------------|
| <b>pre</b> | <b>stkupd</b> | <b>post</b> |
|------------|---------------|-------------|

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

|               |             |
|---------------|-------------|
| <b>sccext</b> | <b>post</b> |
|---------------|-------------|



## 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 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 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 Retail Merchandising System Operations Guide.



| RMS,RTM,ReSA Program Dependency and Scheduling Details |                                      |          |                         |          |  |   |               |                       |  |  |
|--|--------------------------------------|----------|-------------------------|----------|--|---|---------------|-----------------------|--|--|
| Program Name   | Functional Area                      | Threaded | Driver                  | Phase    | Pre-dependency   | Post-dependency   | Timing        | Uses Restart/Recovery | Run Parameters for Programs  |  |
| auditprg   | Audit                                | N        | N/A                     | N/A      | N/A  | N/A   | daily         | N                     | auditprg userid/passwd   |  |
| auditsys   | Audit                                | N        | N/A                     | ad hoc   | N/A  | N/A   | daily         | N                     | auditsys userid/passwd   |  |
| batch_allcotsupd.ksh                                   | Cost Component Updates               | Y        | Allocation and Transfer | 2        | N/A  | prepost batch_costcompupd pre elcexcpgr                                   | daily         | N                     | batch_allcotsupd.ksh [-p <# parallel threads>] <connect>   |  |
| batch_costcompupd                                      | Cost Component Updates               | Y        | N/A                     | 2        | N/A  | N/A   | daily         | N                     | <# parallel threads> is the number of threads to run in parallel. The default is the value on RESTART_CONTROL_NUM_THREADS. |  |
| batch_depchrgupd.ksh                                   | Cost Component Updates               | N        | N/A                     | 2        | N/A  | prepost batch_costcompupd pre elcexcpgr                                   | daily         | N                     | prepost userid/passwd batch_costcompupd pr   |  |
| batch_expprofupd.ksh                                   | Cost Component Updates               | N        | N/A                     | 2        | N/A  | prepost batch_costcompupd pre elcexcpgr                                   | daily         | N                     | batch_depchrgupd.ksh <connect>   |  |
| batch_itmcostcompupd.ksh                               | Cost Component Updates               | N        | Location, Supplier      | 2        | N/A  | prepost batch_costcompupd pre elcexcpgr                                   | daily         | N                     | batch_expprofupd.ksh <connect>   |  |
| batch_ordcostcompupd.ksh                               | Cost Component Updates               | Y        | Order                   | 2        | prepost batch_ordcostcompupd pre   | prepost batch_ordcostcompupd post prepost batch_costcompupd pre elcexcpgr | daily         | N                     | batch_itmcostcompupd.ksh [-p <# parallel threads>] <connect>   |  |
| batch_orpos_extract.ksh                                | Point of Sale Interface              | Y        | Store                   | 4        | If RPM pricing info is reqd then run after extraction script                             | prepost batch_ordcostcompupd post prepost batch_costcompupd pre elcexcpgr | daily         | N                     | <# parallel threads> is the number of threads to run in parallel. The default is the value on RESTART_CONTROL_NUM_THREADS. |  |
| ccprg  | Costing                              | N        | N/A                     | ad hoc   | RPMTORORPOSPublishExport.sh'   | prepost batch_ordcostcompupd post prepost batch_costcompupd pre elcexcpgr | daily         | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| cednid   | Trade Management                     | Y        | Broker                  | 2        | N/A  | posndid (only if generic POS extract is used)                             | monthly       | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| cmpprg   | Pricing                              | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| cmpupld  | Pricing                              | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| cntrmain   | Contracting                          | N        | N/A                     | 0        | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| cntrordb   | Contracting                          | Y        | Contract                | 3        | rpldj  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| cntrprss   | Contracting                          | Y        | Dept                    | 3        | rpext  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| costeventprg.pc  | Real Time Costing - RMS Impacts      | Y        | Event Type              | 0        | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| cremhierdy   | Reclassification                     | N        | N/A                     | 4        | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dealact  | Deals                                | Y        | Deal Id                 | 3        | prepost dealact_nor pre  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dealcfs  | Deals                                | N        | N/A                     | 3        | prepost dealact_po pre   | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dealdy   | Deals                                | Y        | Location                | 3        | prepost dealact_sales pre  | posndid (only if generic POS coupon extract is used)                      | monthly       | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dealfct  | Deals                                | Y        | Deal Id                 | 3        | prepost dealact_pos  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dealfnc  | Deals                                | Y        | Deal Id                 | 3        | prepost dealact_salnmth  | posndid (only if generic POS coupon extract is used)                      | weekly/ad hoc | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dealinc  | Deals                                | Y        | Deal Id                 | 3        | prepost dealact_salnmth  | posndid (only if generic POS coupon extract is used)                      | monthly       | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dealprg  | Deals                                | N        | N/A                     | ad hoc   | prepost dealact_salnmth  | posndid (only if generic POS coupon extract is used)                      | monthly       | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dealupld   | Deals                                | Y        | File-based              | 0        | prepost dealact_salnmth  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dftbld   | Item Maintenance                     | Y        | Dept                    | 3        | (This program is the first one in Deals batch  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| disotbaply   | OTB                                  | Y        | Dept                    | 4        | (This program will likely be run after sales information is uploaded into Oracle Retail) | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| distrocpub   | Pricing/Transfers/Allocation Publish | Y        | Store                   | 3        | ordscnt  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dtinsrt  | Deals                                | N        | N/A                     | 1        | PriceEventExecutionBatch(RPM)  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dyprg  | Maintenance                          | N        | N/A                     | 0        | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dcclose  | Receiving                            | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dsldnid  | Replenishment                        | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dtesys   | Calendar                             | N        | N/A                     | date_set | (This program should run at the end of the batch cycle)                                  | posndid (only if generic POS coupon extract is used)                      | daily         | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| dummycyn   | Receiving                            | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edidfld  | Maintenance                          | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edidcon  | Contracting                          | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edidlinv   | Invoice Matching                     | Y        | Location                | 4        | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edidord  | Ordering                             | N        | N/A                     | 4        | ordrev   | posndid (only if generic POS coupon extract is used)                      | ad hoc        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edidprd  | EDI Interface - Sales and Inventory  | N        | N/A                     | 4        | (and after replenishment)  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| ediprg   | EDI Interface - Purge                | N        | N/A                     | ad hoc   | (Towards the end of the batch cycle)   | posndid (only if generic POS coupon extract is used)                      | monthly       | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edupadd  | Maintenance                          | N        | File-based              | 2        | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edupack  | EDI Interface - ordering             | N        | N/A                     | 1        | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edupavl  | EDI Interface - Contracts            | N        | File-based              | 1        | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| edupcat  | EDI Interface - Suppliers            | N        | File-based              | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| elcexcpgr  | Cost Component Updates               | N        | N/A                     | 2        | N/A  | posndid (only if generic POS coupon extract is used)                      | daily         | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| fcexec   | Real Time Costing                    | Y        | Cost Event Process Id   | 2        | prepost fcexec pre   | posndid (only if generic POS coupon extract is used)                      | daily/ad hoc  | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| fcstprg  | Forecasting                          | Y        | Domain Id               | 3        | prepost fcstprg pre  | posndid (only if generic POS coupon extract is used)                      | weekly        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| fcstbrld   | Forecasting                          | Y        | Domain Id               | 3        | prepost fcstbrld post  | posndid (only if generic POS coupon extract is used)                      | weekly        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| fcstbrld_sbc   | Forecasting                          | Y        | Domain Id               | 3        | prepost fcstbrld post  | posndid (only if generic POS coupon extract is used)                      | weekly        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| fflgdn1  | Financial Interface                  | Y        | Dept                    | 3        | salstage   | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| fflgdn2  | Financial Interface                  | Y        | Dept                    | 3        | salstage   | posndid (only if generic POS coupon extract is used)                      | daily         | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| fflgdn3  | Financial Interface                  | Y        | Store/Wh                | 3        | salnmth  | posndid (only if generic POS coupon extract is used)                      | monthly       | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| ftmednid   | Planning System Interface            | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| gcupld   | Misc Interface - Taxgeocode          | N        | N/A                     | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| genprss  | Ordering                             | Y        | Supplier                | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| gradupld   | Forecasting                          | N        | File-based              | ad hoc   | N/A  | posndid (only if generic POS coupon extract is used)                      | ad hoc        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| hstbld   | Sales                                | Y        | Location                | 3        | prepost hstbld pre (for rebuild all)   | posndid (only if generic POS coupon extract is used)                      | weekly        | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| hstbld_diff  | Sales                                | N        | N/A                     | ad hoc   | hstbld   | posndid (only if generic POS coupon extract is used)                      | ad hoc        | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| hstbldmth  | Sales                                | Y        | Dept                    | 3        | prepost hstbldmth post   | posndid (only if generic POS coupon extract is used)                      | monthly       | R                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |
| hstbldmth_diff   | Sales                                | N        | N/A                     | ad hoc   | prepost hstbldmth post   | posndid (only if generic POS coupon extract is used)                      | ad hoc        | N                     | ch_ordcostcompupd.ksh [-p <# parallel threads>] <connect>  |  |

|                 |                           |   |                  |            |   |  |              |   |   |
|-----------------|---------------------------|---|------------------|------------|---|--|--------------|---|---|
| hstmthupd       | Sales                     | Y | Location         | 3          | (The program should be run on the last day of the month). | (Run SQL*Loader using the control file hstmthupdctl to load data from the output file written by HSTMTHUPD.PC for non-existent records on ITEM_LOC_HIST_MTH) | monthly      | R | hstmthupd userid/passwd (out_file)  |
| hstprg          | Sales                     | N | N/A              | ad hoc     | N/A   | N/A  | monthly      | N | hstprg userid/passwd  |
| hstprg_diff     | Sales                     | N | N/A              | ad hoc     | N/A   | Run SQL*Loader using the control file hstwkupd.ctf to load data from the output file written by HSTWKUPD.PC for non-existent records on ITEM_LOC_HIST        | weekly       | N | hstprg_diff userid/passwd   |
| hstwkupd        | Sales                     | Y | Store/Wh         | 3          | N/A   | Hts240_to_2400 (perl script)   | weekly       | R | hstwkupd userid/passwd (out_file)   |
| hstsupd         | Trade Management          | Y | File-based       | ad hoc     | Ushs2rms (perl script)                                    | prepost hstsupd pre  | ad hoc       | R | hstsupd userid/passwd input_file reject_file country_id ; perl hts_240_to_2400 inputfile outputfile ; perl ushs2rms inputfile outputfile rejectfile                               |
| ibcalc          | Investment Buy            | Y | Dept             | 3          | prepost ibcalc pre  | rbtbd  | daily        | R | ibcalc userid/passwd  |
| ibexpl          | Investment Buy            | N | N/A              | ad hoc     | rplext  | ibcalc   | daily        | N | ibexpl userid/passwd  |
| invaprg         | Inventory Adjustments     | N | N/A              | ad hoc     | N/A   | N/A  | monthly      | N | invaprg userid/passwd   |
| invclshp        | Invoice Matching          | N | N/A              | 2          | N/A   | N/A  | daily        | N | invclshp userid/passwd  |
| invprg          | Invoice Matching          | N | N/A              | ad hoc     | ordprg  | N/A  | monthly      | R | invprg userid/passwd  |
| lcardnd         | Letter of Credit          | N | N/A              | 4          | N/A   | lcm700 (perl script)   | daily        | R | lcardnd userid/passwd output_file   |
| lclrbid         | Maintenance - Location    | N | N/A              | ad hoc     | storeadd  | N/A  | monthly      | R | lclrbid userid/passwd   |
| lcmndnd         | Letter of Credit          | N | N/A              | 4          | N/A   | lcm707 (perl script)   | daily        | R | lcmndnd userid/passwd output_file   |
| lcup798         | Letter of Credit          | N | N/A              | 2          | lcm798 (perl script)                                      | N/A  | daily        | R | lcup798 userid/passwd input_file rej_fil  |
| lcupld          | Letter of Credit          | N | N/A              | 2          | lcm730 (perl script)                                      | N/A  | daily        | R | lcupld userid/passwd input_file rej_fil   |
| lftskup         | Stock Ledger              | N | File-based       | 1          | inv_bal_upload.sh (warehouse mgmt program)                | stkupld  | daily        | N | lftskup userid/passwd input_file output_file  |
| lftstore        | Maintenance - Location    | Y | Dept             | ad hoc     | storeadd  | prepost lftstore pos mtrtrv  | daily        | R | lftstore userid/passwd  |
| mrt             | Mass Return Transfers     | Y | Warehouse        | 2          | N/A   | mrtupd   | daily        | R | mrt userid/passwd   |
| mrtprg          | Mass Return Transfers     | Y | Warehouse        | ad hoc     | N/A   | mrtupd   | ad hoc       | R | mrtprg userid/passwd  |
| mtrtrv          | Mass Return Transfers     | Y | Warehouse        | 2          | mrt   |  | daily        | R | mtrtrv userid/passwd  |
| mrtupd          | Mass Return Transfers     | Y | Warehouse        | 2          | mtrtrv  | N/A  | daily        | R | mrtupd userid/passwd  |
| nwppurge        | Stock Ledger              | N | N/A              | ad hoc     | N/A   | N/A  | ad hoc       | N | nwppurge userid/passwd  |
| nwpyearend      | Stock Count               | Y | Location         | 4          | run on last day of yea                                    | N/A  | yearly       | R | nwpyearend userid/passwd  |
| ociroq          | Replenishment             | N | N/A              | 3/Ad hoc   | prepost ociroq pre repladj                                | N/A  | daily/adhoc  | R | Y- for nightly batch run.   |
| onictext        | Planning System Interface | Y | Transfer         | 4          | onordext  | onordnd  | daily        | R | N- for multiple (ad hoc run)  |
| onordnd         | Planning System Interface | Y | Store/Wh         | 4          | onictext  | N/A  | daily        | R | onictext userid/passwd datefil  |
| onordext        | Planning System Interface | Y | Order            | 4          | prepost onordext pri                                      | onictext   | daily        | R | onordnd userid/passwd   |
| ordautcl        | Ordering                  | N | N/A              | ad hoc     | N/A   | N/A  | daily        | N | onordext userid/passwd datefil  |
| orddsent        | Deals                     | Y | Supplier         | 4          | discothapply  | discothapply   | daily        | R | ordautcl userid/passwd  |
| ordprg          | Ordering                  | N | N/A              | ad hoc     | recdsdy   | dealcis  | daily        | R | orddsent userid/passwd  |
| ordrev          | Ordering                  | N | N/A              | 4          | orddsent  | invprg   | monthly      | N | ordprg userid/passwd  |
| ordupd          | Ordering                  | N | N/A              | 4          | orddsent  | edidord  | daily        | R | ordrev userid/passwd  |
| otbdord         | Ordering                  | N | N/A              | 4          | otbdord   | otbdord  | daily        | N | ordupd userid/passwd  |
| otbdial         | OTB                       | N | N/A              | 4          | otbdord   | otbdord  | daily        | R | otbdord userid/passwd output_file   |
| otbdnd          | OTB                       | N | N/A              | 4          | otbdord   | otbdord  | daily        | R | otbdial userid/passwd output_file   |
| otbprg          | OTB                       | N | N/A              | ad hoc     | N/A   | N/A  | monthly      | N | otbdnd userid/passwd output_fil   |
| otbupfwd        | OTB                       | Y | File-based       | ad hoc     | N/A   | N/A  | daily        | R | otbprg userid/passwd  |
| otbupld         | OTB                       | Y | File-based       | ad hoc     | N/A   | N/A  | daily        | R | userid/passwd input_file reject_fil   |
| poscdnd         | Point of Sale Interface   | N | N/A              | 4          | poscdnd   | prepost poscdnd post   | daily        | R | otbupld userid/passwd input_file reject_fil   |
| posdnd          | Point of Sale Interface   | Y | Store            | ad hoc     | N/A   | prepost posdnd post  | daily        | R | poscdnd userid/passwd outputfil   |
| posgpdld        | Point of Sale Interface   | N | N/A              | 4          | recdsdy   | N/A  | daily        | R | posdnd userid/passwd output_filename  |
| posupld         | Sales                     | Y | File-based       | 2          | saaxprms(ReSA)  | prepost posupld post   | daily        | R | posgpdld userid/passwd output_file  |
| prepost         | Pre/post functionality    | N | N/A              | all phases | N/A   | salstage   | daily        | R | posupld userid/passwd infile rejfile vattile itemfile lockfile  |
| recdsdy         | Item Maintenance          | Y | Reclass no       | 4          | cremhierdly   | N/A  | daily        | N | prepost userid/passwd program pre_or_pos  |
| repladj         | Replenishment             | Y | Dept             | 3          | prepost recdsdy pre                                       | prepost recdsdy post   | daily        | R | recdsdy userid/passwd process_mode  |
| replsizeprofile | Replenishment             | N | N/A              | ad hoc     | repladj   | repladj  | daily        | R | repladj userid/passwd   |
| reqlt           | Replenishment             | Y | Partition (Item) | 3/Ad hoc   | prepost ociroq pre ociroq                                 | prepost reqlt post   | daily/ad hoc | R | replsizeprofile userid/passwd Y/N, (Y/N indicator indicates if allocations is installed or not, if installed pre job for this program has to be run prepost replsizeprofile pre)  |
| rlmaint         | Replenishment             | Y | Location         | 3          | reqlt   | reqlt  | daily        | R | reqlt userid/passwd partition_position last_run_of_the_day_ind(Y/N) (May use the batch_reqlt.kah for launching this program as it is created based on performance considerations) |
| rlapprv         | Replenishment             | N | N/A              | 3          | supcnstr  | supcnstr   | daily/adhoc  | R | Y- for nightly batch run.   |
| rlathistprg     | Replenishment             | N | N/A              | ad hoc     | N/A   | N/A  | ad hoc       | N | N- for multiple (ad hoc run)  |
| rlatupd         | Replenishment             | Y | Location         | 3          | reqlt   | reqlt  | daily        | R | rlmaint username/password   |
| rbtbd           | Replenishment             | Y | Supplier         | 3/Ad hoc   | ibexpl  | supcnstr   | daily/ad hoc | R | rlapprv userid/passwd   |



|          |                         |   |            |        |   |                       |         |   |   |
|----------|-------------------------|---|------------|--------|---|-----------------------|---------|---|---|
| tsfprg   | Transfers               | N | N/A        | ad hoc | N/A   | N/A                   | monthly | R | tsfprg userid/passwd                              |
| txrposdn | Point of Sale Interface | N | N/A        | 4      | N/A   | txrposdn              | daily   | R | txrposdn userid/passwd                            |
| txrtupld | Sales Tax               | N | N/A        | 4      | N/A   | N/A                   | ad hoc  | R | txrtupld username/password input_file reject_file |
| vatdxpl  | Maintenance - VAT       | Y | Vat Region | 0      | N/A   | prepost vatdxpl pos   | daily   | R | vatdxpl userid/passwd                             |
| vendinvc |                         |   |            |        | dealact   | prepost vendinvc post |         |   |   |
|          | Deals                   | Y | Deal Id    | 3      | salstage(if daily)                                    | salweek(if weekly)    | daily   | R | vendinvc userid/passwd                            |
|          |                         |   |            |        | prepost vendinvc pre                                  | salmonth (if monthly) |         |   |   |
| vendinvf |                         |   |            |        | prepost vendinvf post                                 | salweek(if weekly)    |         |   |   |
|          | Deals                   | Y | Deal Id    | 3      | salstage(if daily)                                    | salmonth (if monthly) | daily   | R | vendinvf userid/passwd                            |
|          | Replenishment           | Y | Supplier   | 2      | prepost vendinvf pre                                  | edupack               | daily   | R | vrplbld userid/passwd                             |
| vrplbld  |                         |   |            |        |   | prepost vrplbld post  |         |   |   |
| wasteadj | Stock Ledger            | Y | Store      | 3      | N/A   | stkupld               | daily   | R | wasteadj userid/passwd                            |
| whadd    | Maintenance - Location  | N | N/A        | ad hoc | N/A   | N/A                   | daily   | R | whadd userid/passwd                               |
| whstrasg | Maintenance - Location  | N | N/A        | 3      | (Must be run after all replenishment batch programs). | prepost whstrasg post | daily   | R | whstrasg userid/passwd                            |

| RPM Dependency and Scheduling Details         |                                  |          |                        |       |                                   |                                   |              |                       |   |
|---|----------------------------------|----------|------------------------|-------|-----------------------------------|-----------------------------------|--------------|-----------------------|---|
| Program Name                                  | Functional Area                  | Threaded | Driver                 | Phase | Pre-dependency                    | Post-dependency                   | Timing       | Uses Restart/Recovery | Run Parameters for Programs   |
| ItemReclassBatch                              | Future Retail                    | N        | N/A                    | N/A   | recsdy(RMS)                       | NewItemLocBatch                   | daily/ad hoc | N                     | ItemReclassBatch.sh rpm-app-userid password   |
| NewItemLocBatch                               | Future Retail                    | N        | N/A                    | N/A   | storeadd(RMS), ItemReclassBatch   | LocationMoveScheduleBatch         | daily/ad hoc | N                     | NewItemLocBatch.sh rpm-app-userid password [status [error-commit-count]                                     |
| LocationMoveScheduleBatch                     | Zone Structure/Future Retail     | Y        | N/A                    | N/A   | NewItemLocBatch                   | LocationMoveBatch                 | daily        | N                     | locationMoveScheduleBatch userid password   |
| LocationMoveBatch                             | Zone Structure/Future Retail     | Y        | Location move          | N/A   | LocationMoveScheduleBatch         | PriceEventExecutionBatch          | daily        | N                     | locationMoveBatch.sh rpm-app-userid password  |
| PriceEventExecutionBatch                      | Price Change/Clearance/Promotion | Y        | Pricing event          | N/A   | LocationMoveBatch                 | PriceEventExecutionRMSBatch       | daily        | N                     | priceEventExecutionBatch.sh rpm-app-userid password   |
| PriceEventExecutionRMSBatch                   | Price Change/Clearance/Promotion | Y        | Pricing event          | N/A   | PriceEventExecutionBatch          | PriceEventExecutionDealsBatch     | daily        | N                     | priceEventExecutionRMSBatch.sh rpm-app-userid password  |
| PriceEventExecutionDealsBatch                 | Price Change/Clearance/Promotion | Y        | Pricing event          | N/A   | PriceEventExecutionRMSBatch       | MerchExtractKickOffBatch          | daily        | N                     | priceEventExecutionDealsBatch.sh rpm-app-userid password  |
| PriceStrategyCalendarBatch                    | Price Strategy                   | N        | Price strategy         | N/A   | N/A                               | MerchExtractKickOffBatch          | daily        | N                     | priceStrategyCalendarBatch.sh rpm-app-userid password   |
| WorksheetAutoApproveBatch                     | Pricing Worksheet                | Y        | Price strategy         | N/A   | PriceEventExecutionBatch          | MerchExtractKickOffBatch          | daily        | N                     | worksheetAutoApproveBatch.sh rpm-app-userid password  |
| MerchExtractKickOffBatch                      | Pricing Worksheet                | Y        | Price strategy         | N/A   | PriceEventExecutionBatch          | N/A                               | daily        | N                     | merchExtractKickOffBatch.sh rpm-app-userid password   |
| RPMTORPOSPublishBatch.sh                      | Price Change/Clearance/Promotion | N        | N/A                    | N/A   | WorksheetAutoApproveBatch         | N/A                               | daily        | N                     | ksh RPMTORPOSPublishBatch.sh -userid/password@sid > <log path> <error path>                                 |
| RPMTORPOSPublishExport.sh                     | Price Change/Clearance/Promotion | Y        | Location               | N/A   | RPMTORPOSPublishBatch.sh          | N/A                               | daily        | N                     | ksh RPMTORPOSPublishExport.sh -userid/password@sid > <Number of slots> <logpath> <error path> <Export path> |
| RegularPriceChangePublishBatch                | Regular Price Changes            | Y        | Price event (item/loc) | N/A   | WorksheetAutoApproveBatch         | RegularPriceChangePublishExport   | daily/ad hoc | N                     | regularPriceChangePublishBatch.sh rpm-app-userid password   |
| RegularPriceChangePublishExport               | Regular Price Changes            | N        | Price event (item/loc) | N/A   | RegularPriceChangePublishBatch    | N/A                               | daily/ad hoc | N                     | regularPriceChangePublishExport.sh rpm-db-userid/pwd@database [export-path]                                 |
| ClearancePriceChangePublishBatch              | Clearances                       | Y        | Price event (item/loc) | N/A   | WorksheetAutoApproveBatch         | ClearancePriceChangePublishExport | daily/ad hoc | N                     | clearancePriceChangePublishBatch.sh rpm-app-userid password   |
| ClearancePriceChangePublishExport             | Clearances                       | N        | Price event (item/loc) | N/A   | ClearancePriceChangePublishBatch  | N/A                               | daily/ad hoc | N                     | clearancePriceChangePublishExport.sh rpm-db-userid/pwd@database [export-path]                               |
| PromotionPriceChangePublishBatch              | Promotions                       | Y        | Price event (item/loc) | N/A   | WorksheetAutoApproveBatch         | PromotionPriceChangePublishExport | daily/ad hoc | N                     | promotionPriceChangePublishBatch.sh rpm-app-userid password   |
| PromotionPriceChangePublishExport             | Promotions                       | N        | Price event (item/loc) | N/A   | PromotionPriceChangePublishBatch  | N/A                               | daily/ad hoc | N                     | promotionPriceChangePublishExport.sh rpm-db-userid/pwd@database [export-path]                               |
| PriceChangeAutoApproveResultsPurgeBatch       | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | daily/ad hoc | N                     | priceChangeAutoApproveResultsPurgeBatch.sh rpm-app-userid password  |
| PriceChangePurgeBatch                         | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | daily/ad hoc | N                     | priceChangePurgeBatch.sh rpm-app-userid password  |
| PriceChangePurgeWorkspaceBatch                | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | daily/ad hoc | N                     | priceChangePurgeWorkspaceBatch.sh rpm-app-userid password   |
| PromotionPurgeBatch                           | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | daily/ad hoc | N                     | promotionPurgeBatch.sh rpm-app-userid password  |
| PurgeExpiredExecutedOrApprovedClearancesBatch | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | daily/ad hoc | N                     | purgeExpiredExecutedOrApprovedClearancesBatch.sh rpm-app-userid password                                    |
| PurgeUnusedAndAbandonedClearancesBatch        | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | daily/ad hoc | N                     | purgeUnusedAndAbandonedClearancesBatch.sh rpm-app-userid password   |
| PurgeLocationMovesBatch                       | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | daily/ad hoc | N                     | purgeLocationMovesBatch.sh rpm-app-userid password  |
| ZoneFutureRetailPurgeBatch                    | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | daily        | N                     | zoneFutureRetailPurgeBatch.sh rpm-app-userid password   |
| ItemLocDeleteBatch                            | Purge                            | N        | N/A                    | N/A   | N/A                               | N/A                               | ad hoc       | N                     | itemLocDeleteBatch.sh rpm-app-userid password   |
| priceChangeAreaDifferentialBatch              | Price Change                     | Y        | N/A                    | N/A   | N/A                               | N/A                               | ad hoc       | N                     | priceChangeAreaDifferentialBatch rpm-app-userid password  |
| refreshPosDataBatch.sh                        | Purge                            | Y        | N/A                    | N/A   | N/A                               | N/A                               | ad hoc       | N                     | refreshPosDataBatch -username- <password> -location-[date (YYYYMMdd   |
| injectorPriceEventBatch                       | Price Change/Clearance/Promotion | Y        | N/A                    | N/A   | N/A                               | N/A                               | ad hoc       | N                     | InjectorPriceEventBatch user_name password status=status_value event_type=event_valu                        |
|   |                                  |          |                        |       |                                   |                                   |              |                       | polling_interval=polling_interval_value   |
| purgePayloadBatch                             | Purge                            | N        | N/A                    | N/A   | RPMTORPOSPublishExport            | N/A                               | ad hoc       | N                     | purgePayloadBatch.sh -userid/pwd@database> <publish-status>   |
| purgeBulkConflictCheckArtifacts               | Purge                            | N        | N/A                    | N/A   | RegularPriceChangePublishExport   | N/A                               | ad hoc       | N                     | purgeBulkConflictCheckArtifacts userid password   |
| taskPurgeBatch.sh                             | Purge                            | N        | N/A                    | N/A   | PromotionPriceChangePublishExport | N/A                               | daily/ad hoc | N                     | taskPurgeBatch.sh -username> <password> [-purgeDays>] [Y/N]   |

| ReIM Dependency and Scheduling Details |                         |          |        |       |                            |                 |        |                       |   |
|--|-------------------------|----------|--------|-------|----------------------------|-----------------|--------|-----------------------|---|
| Program Name                           | Functional Area         | Threaded | Driver | Phase | Pre-dependency             | Post-dependency | Timing | Uses Restart/Recovery | Run Parameters for Programs   |
| reimautomatch                          | Invoice Matching (ReIM) | Y        | N/A    | 6     | N/A                        | reimrollup      | daily  | R                     | reimautomatch userid/password   |
| reimpurge                              | Invoice Matching (ReIM) | N        | N/A    | 0     | N/A                        | reimautomatch   | daily  | R                     | reimpurge userid/password   |
| reimcomplexdealupload                  | Invoice Matching (ReIM) | Y        | N/A    | 5     | vendinv(RMS), vendinv(RMS) | reimautomatch   | daily  | R                     | reimcomplexdealupload userid/password BlockSize PartitionNo                             |
| reimdiscrepancy                        | Invoice Matching (ReIM) | N        | N/A    | 1     | N/A                        | N/A             | daily  | R                     | reimdiscrepancy userid/password   |
| reimeditinupload                       | Invoice Matching (ReIM) | Y        | N/A    | 5     | eddlinv(RMS)               | reimautomatch   | daily  | R                     | reimeditinupload userid/password "EDI input file with path" "EDI reject file with path" |
| reimeditinupload                       | Invoice Matching (ReIM) | N        | N/A    | 7     | reimposting                | N/A             | daily  | R                     | reimeditinupload userid/password  |
| reimfixdealupload                      | Invoice Matching (ReIM) | Y        | N/A    | 5     | vendinv(RMS), vendinv(RMS) | reimautomatch   | daily  | R                     | reimfixdealupload userid/password BlockSize PartitionNo                                 |
| reimrollup                             | Invoice Matching (ReIM) | N        | N/A    | 6     | reimautomatch              | N/A             | daily  | R                     | reimrollup userid/password  |
| reimreceiptwriteoff                    | Invoice Matching (ReIM) | N        | N/A    | 6     | reimautomatch              | N/A             | daily  | R                     | reimreceiptwriteoff userid/password   |
| reimposting                            | Invoice Matching (ReIM) | N        | N/A    | 6     | reimrollup                 | N/A             | daily  | R                     | reimposting userid/password   |
| reimcreditnotautomatch                 | Invoice Matching (ReIM) | Y        | N/A    | 6     | N/A                        | N/A             | daily  | R                     | reimcreditnotautomatch userid/password  |

| RMS to RPAS RETL Extracts Dependency and Scheduling Details (EXTRACTS FOR RPAS) |                                    |          |        |       |   |                                |        |                       |                             |
|---|------------------------------------|----------|--------|-------|---|--------------------------------|--------|-----------------------|-----------------------------|
| Program Name  | Functional Area                    | Threaded | Driver | Phase | Pre-dependency                                | Post-dependency                | Timing | Uses Restart/Recovery | Run Parameters for Programs |
| pre_rmse_rpas.ksh   | Planning/Forecast System Interface | N        | N/A    | N/A   | N/A. This is a pre setup script               | N/A                            | daily  | N                     | N/A                         |
| rmse_rpas.ksh   | Planning/Forecast System Interface | N        | N/A    | N/A   | pre_rmse_rpas.ksh. (This is the launch script | Refer to RPAS Operations guide | daily  | N                     | N/A                         |
| rmse_rpas_attributes.ksh  | Planning/Forecast System Interface | N        | N/A    | N/A   | to run the extracts)                          | Refer to RPAS Operations guide | daily  | N                     | N/A                         |
| rmse_rpas_daily_sales.ksh   | Planning/Forecast System Interface | N        | N/A    | N/A   | pre_rmse_rpas.ksh                             | Refer to RPAS Operations guide | daily  | N                     | N/A                         |
| rmse_rpas_domain.ksh  | Planning/Forecast System Interface | N        | N/A    | N/A   | pre_rmse_rpas.ksh                             | Refer to RPAS Operations guide | daily  | N                     | N/A                         |
|   |                                    |          |        |       | sitmain                                       |                                |        |                       |                             |
| rmse_rpas_item_master.ksh   | Planning/Forecast System Interface | N        | N/A    | N/A   | recsdy  | Refer to RPAS Operations guide | daily  | N                     | N/A                         |
|   |                                    |          |        |       | recsdy  |                                |        |                       |                             |
| rmse_rpas_merchier.ksh  | Planning/Forecast System Interface | N        | N/A    | N/A   | dyprg   | Refer to RPAS Operations guide | daily  | N                     | N/A                         |
|   |                                    |          |        |       | dyprg   |                                |        |                       |                             |
| rmse_rpas_orghier.ksh   | Planning/Forecast System Interface | N        | N/A    | N/A   | pre_rmse_rpas.ksh                             | Refer to RPAS Operations guide | daily  | N                     | N/A                         |
|   |                                    |          |        |       | stkdy   |                                |        |                       |                             |
| rmse_rpas_stock_on_hand.ksh   | Planning/Forecast System Interface | N        | N/A    | N/A   | pre_rmse_rpas.ksh                             | Refer to RPAS Operations guide | daily  | N                     | N/A                         |

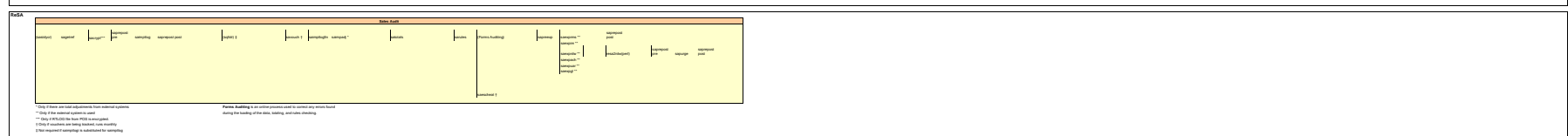
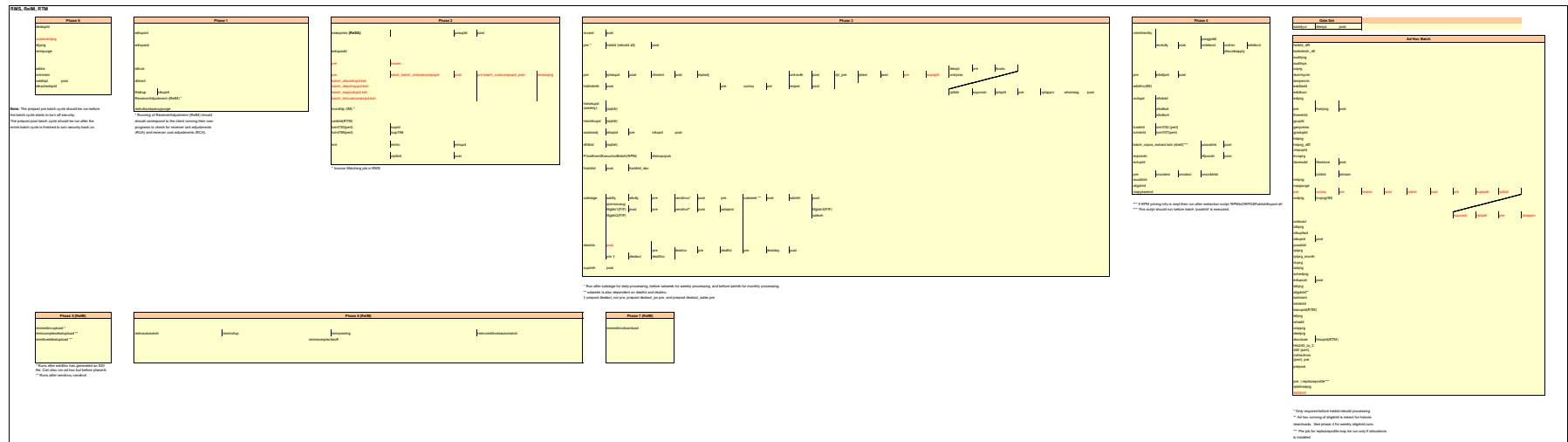








### Integrated Merchandising Batch Schedule





---

## Interface Diagrams for RMS and RPAS

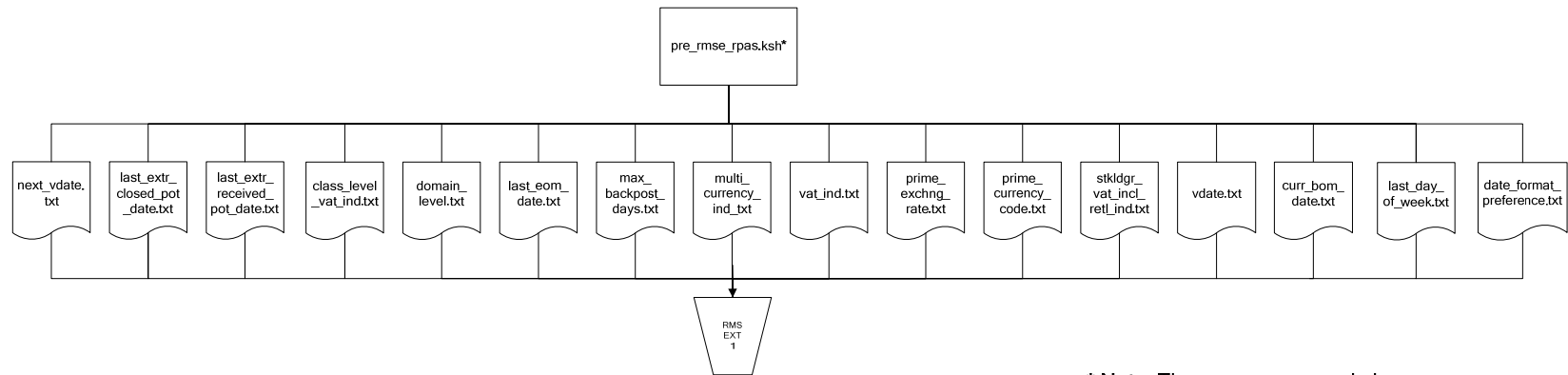
Because RMS is the retailer's central merchandising transactional processing system, it is the principle source of the foundation data needed in some of the Oracle Retail suite of products. RMS provides foundation data to RPAS, and RPAS provides planning data to RMS.

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 Retail Merchandising System Operations Guide for more information about these interface programs.

## RMS Pre/Post Extract Diagrams

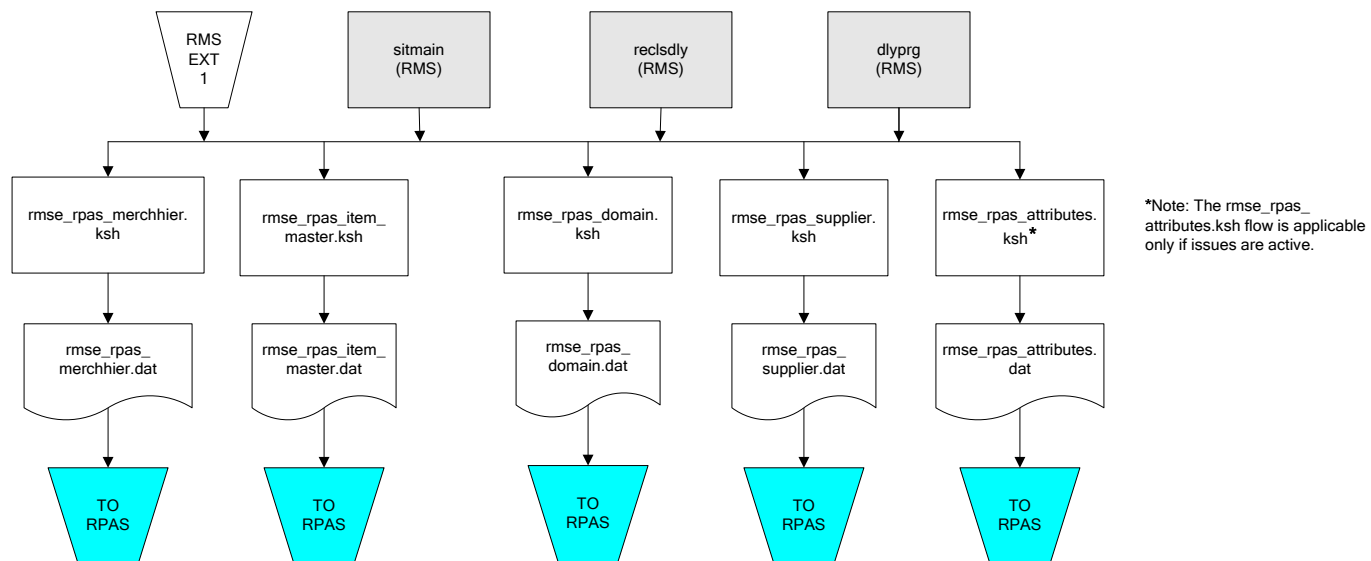
### RMS Pre RETL Extract Maintenance



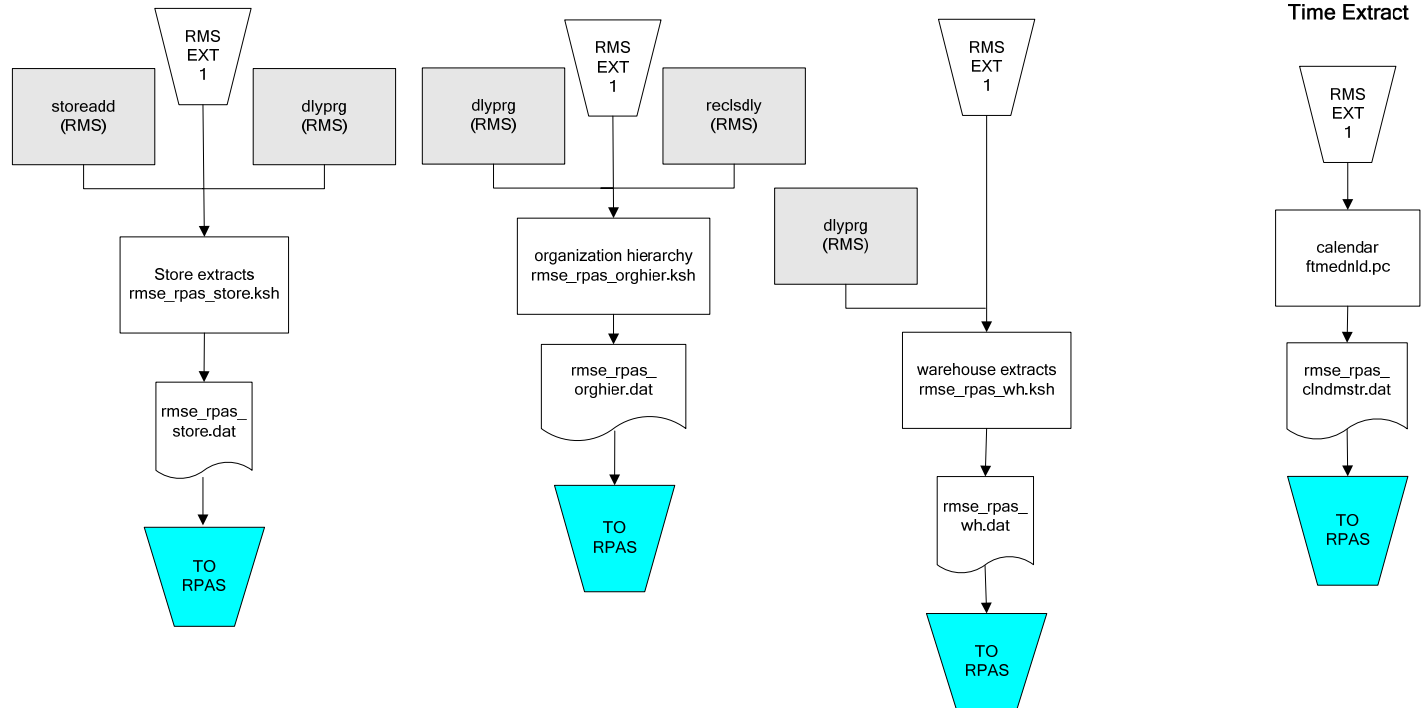
**\* Note:** The `pre_rmse_rpas.ksh` program checks for existing .txt output files. Because of this validation, retailers running the program for the first time should include an optional `-c` parameter. This parameter allows the program to run successfully without pre-existing .txt output files.

## RMS Foundation Data Extract Diagrams

Merchandise Hierarchy for RPAS

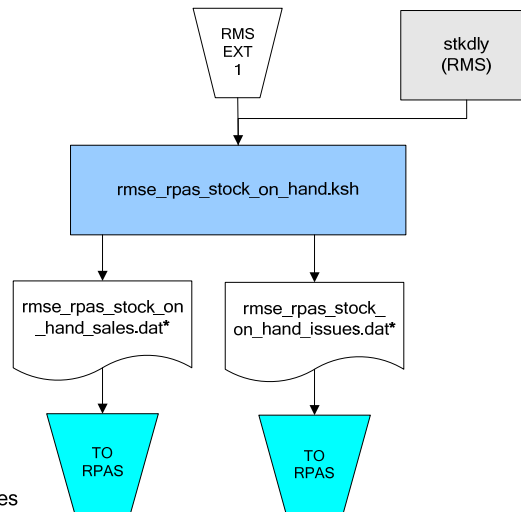


### Organization Hierarchy for RPAS





## RMS Fact Data Extract Diagrams

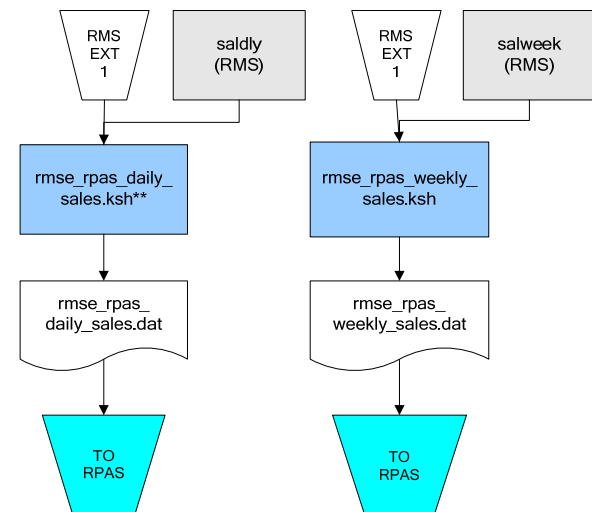


### \* Note:

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

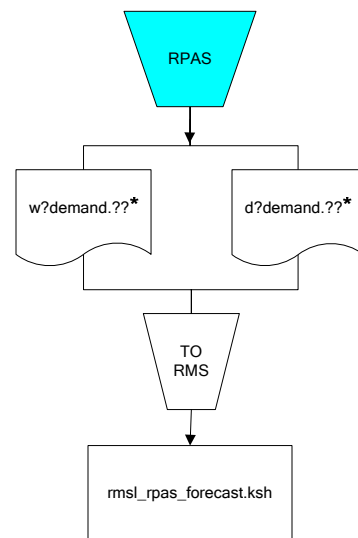
## Sales Extracts For RPAS



### \*\* 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 RDW

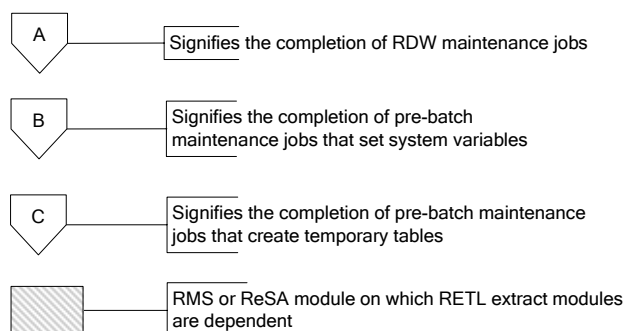
RMS works in conjunction with the Oracle Retail Extract Transform and Load (RETL) framework. RETL provides high-performance processing to extract data from Oracle Retail applications for use in data warehouses. The architecture allows database batch processes to take advantage of parallel processing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

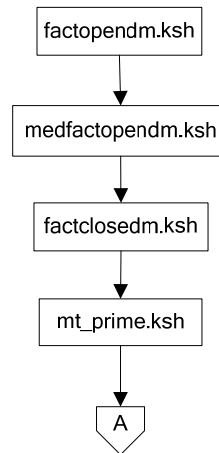
### Legend



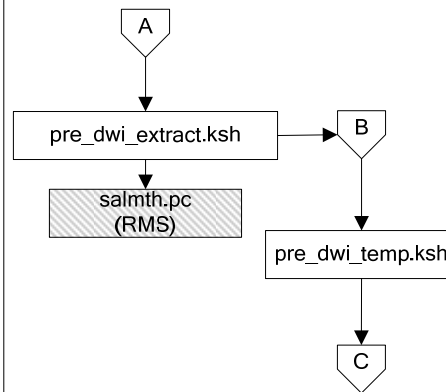
### RDW Maintenance

**Note:**

The modules in this flow are RDW RETL scripts. If the retailer uses RDW, this flow must be completed before starting the pre-batch maintenance flow. If the retailer does not use RDW, these jobs are not required.



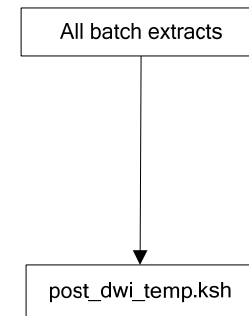
### Pre-Batch Maintenance



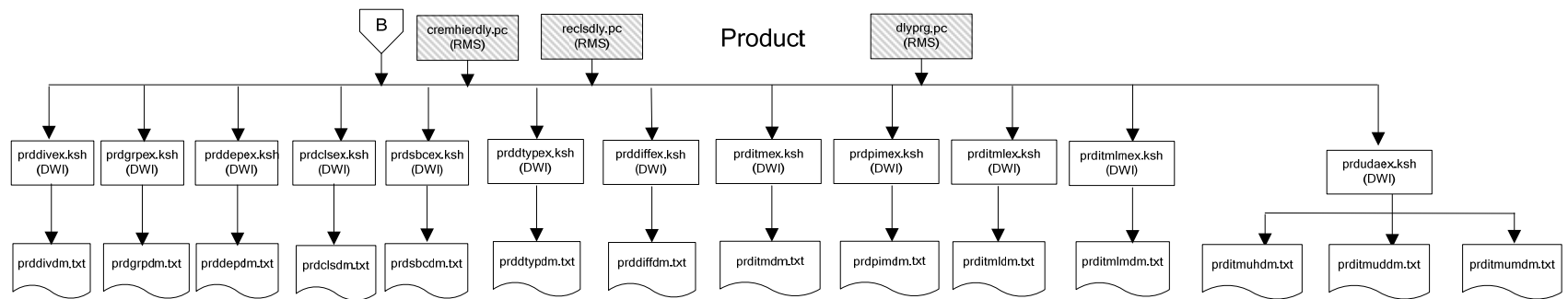
**Note:**

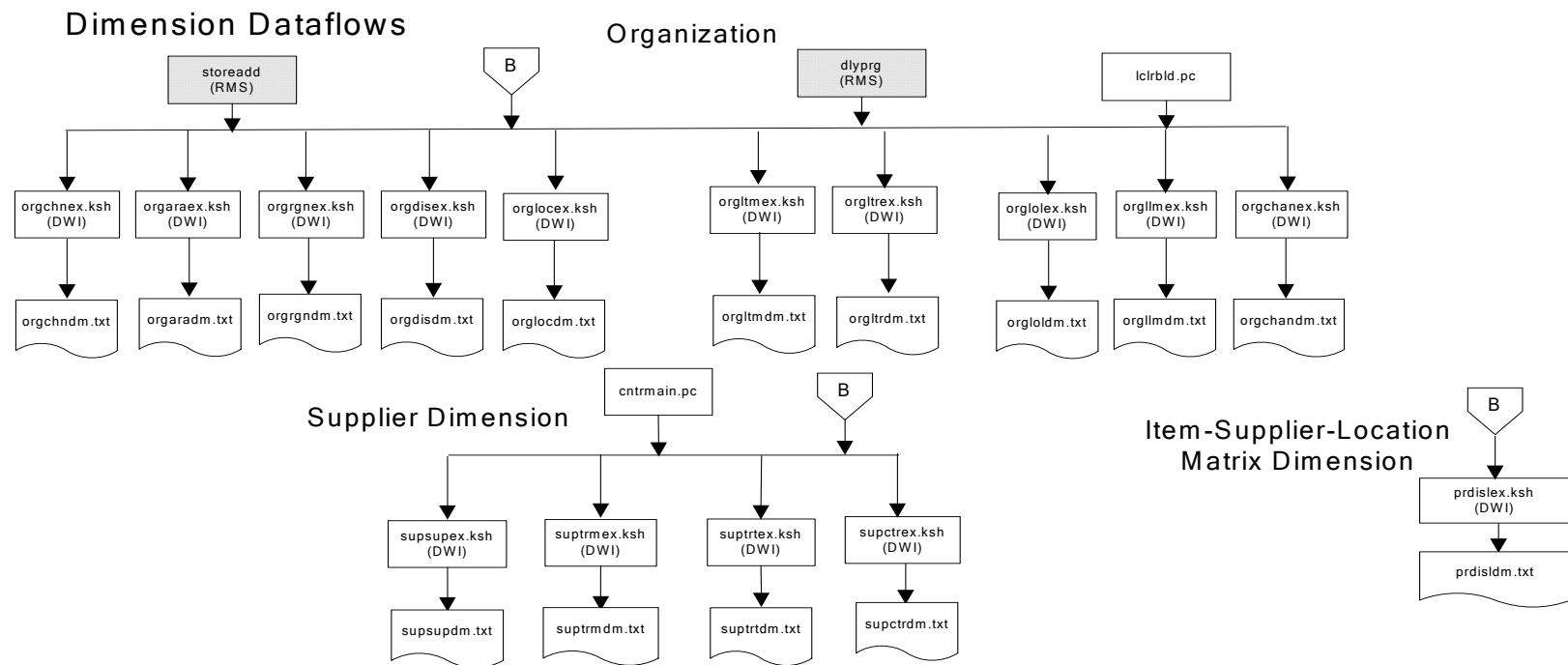
salmth.pc resets the last eom\_date. Thus, it must be run after the system indicator is extracted by pre\_dwi\_extract.ksh.

### Post-Batch Maintenance

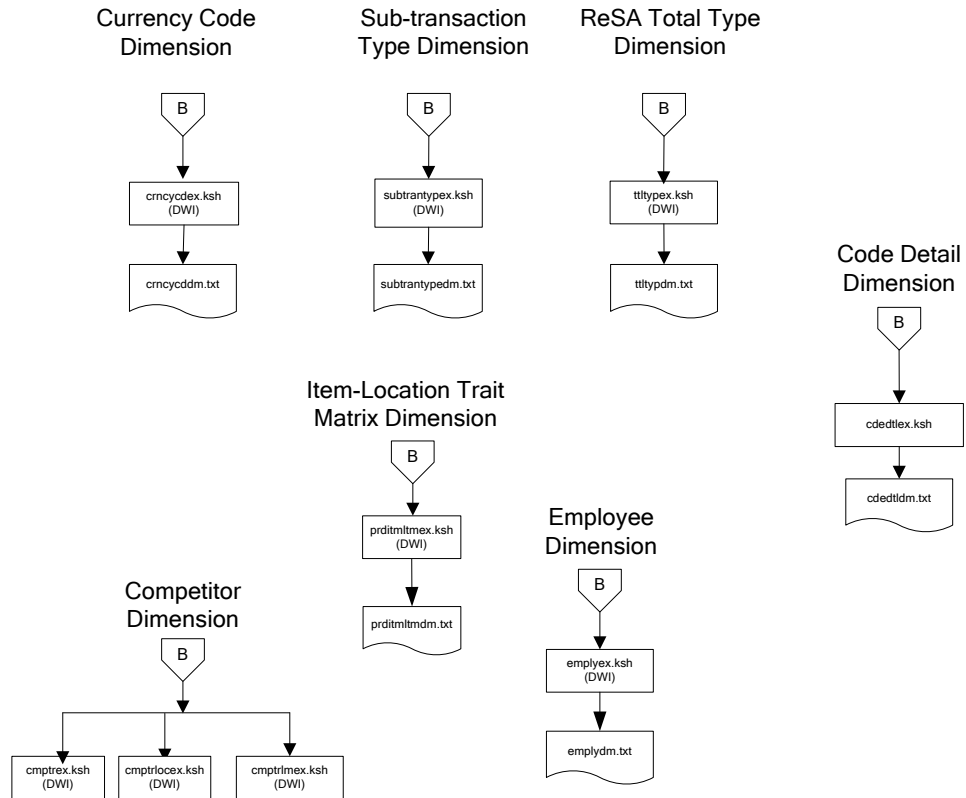


## Dimension Dataflows



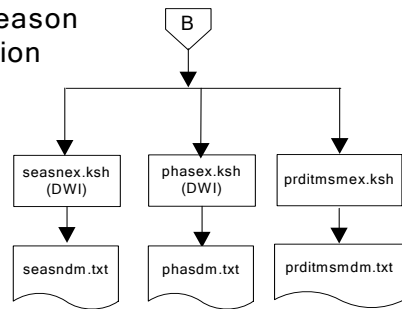


## Dimension Dataflows

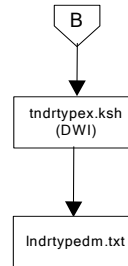


## Dimension Dataflows

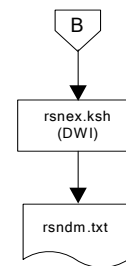
### Product Season Dimension



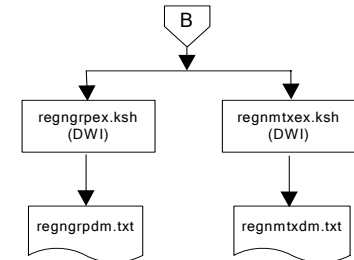
### Tender Type Dimension



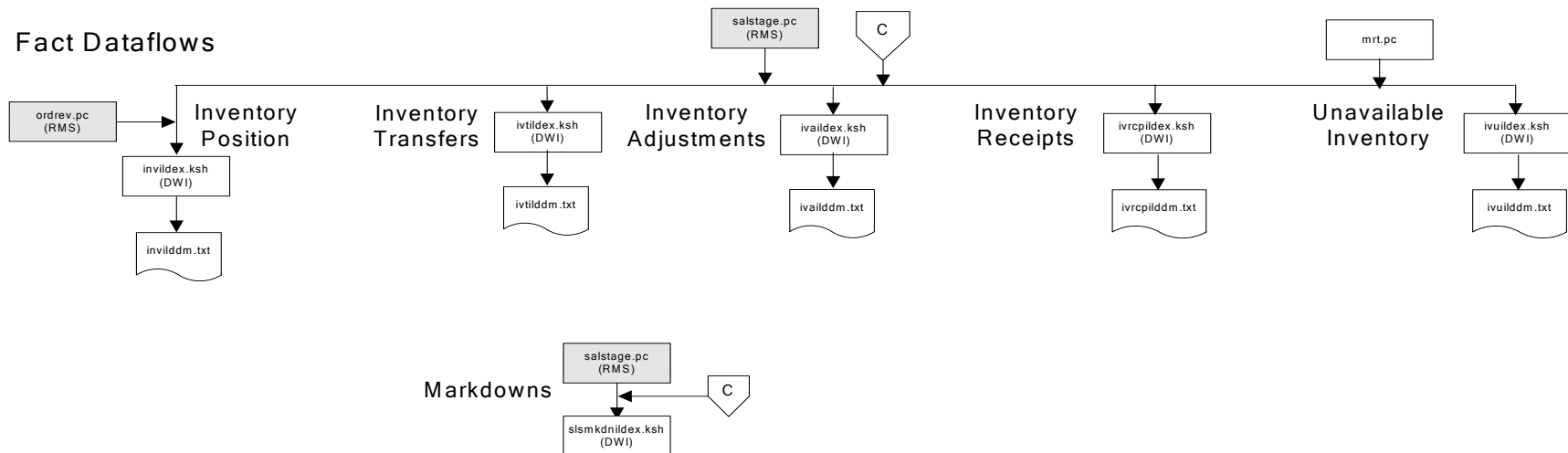
### Reason Dimension



### Regionality Dimension

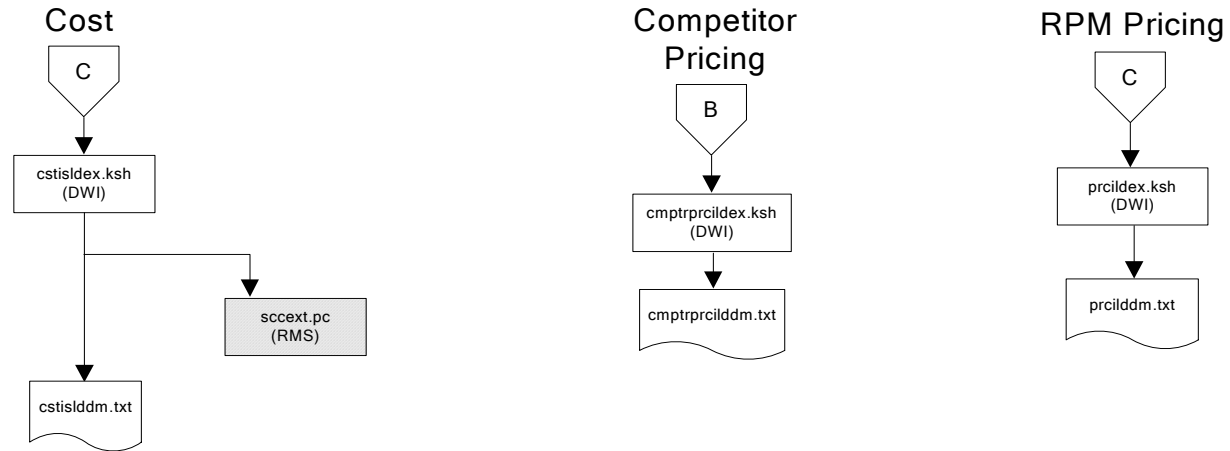


## Fact Dataflows

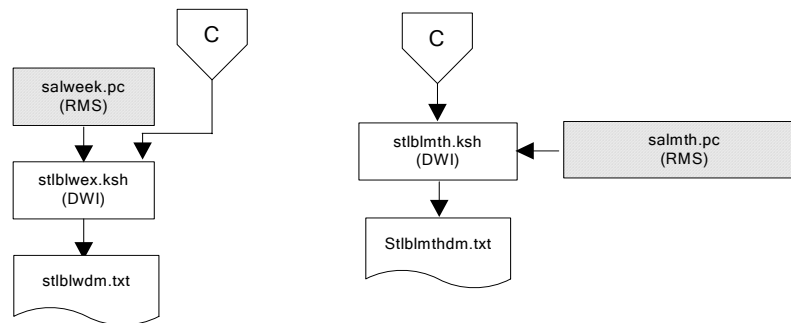




## Fact Dataflows



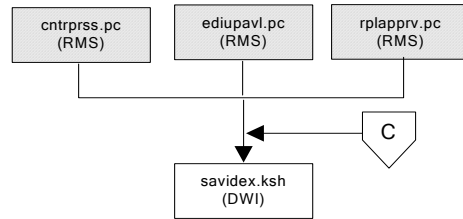
## Stock Ledger



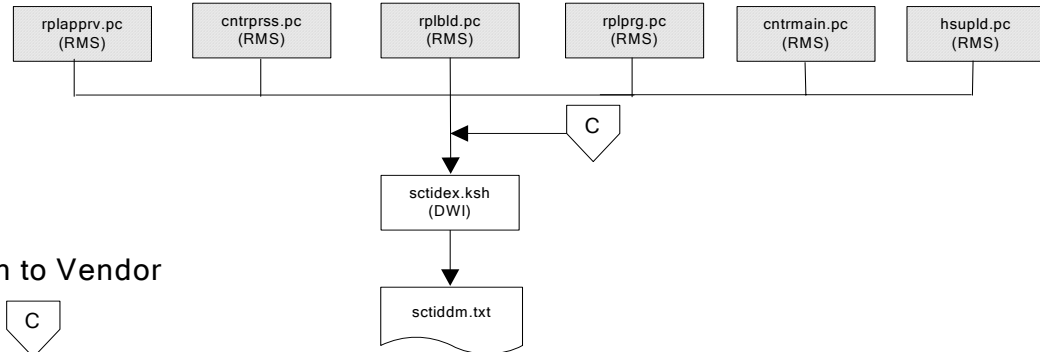
**Note:**  
Run stock ledger fact  
loads once weekly.

## Fact Dataflows

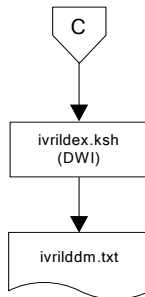
### Supplier Availability



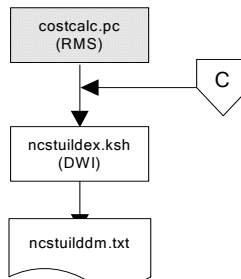
### Supplier Contract



### Return to Vendor

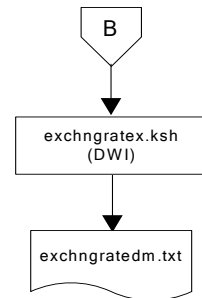


### Net Cost

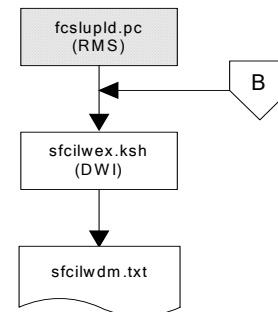


## Fact Dataflows

### Exchange Rates

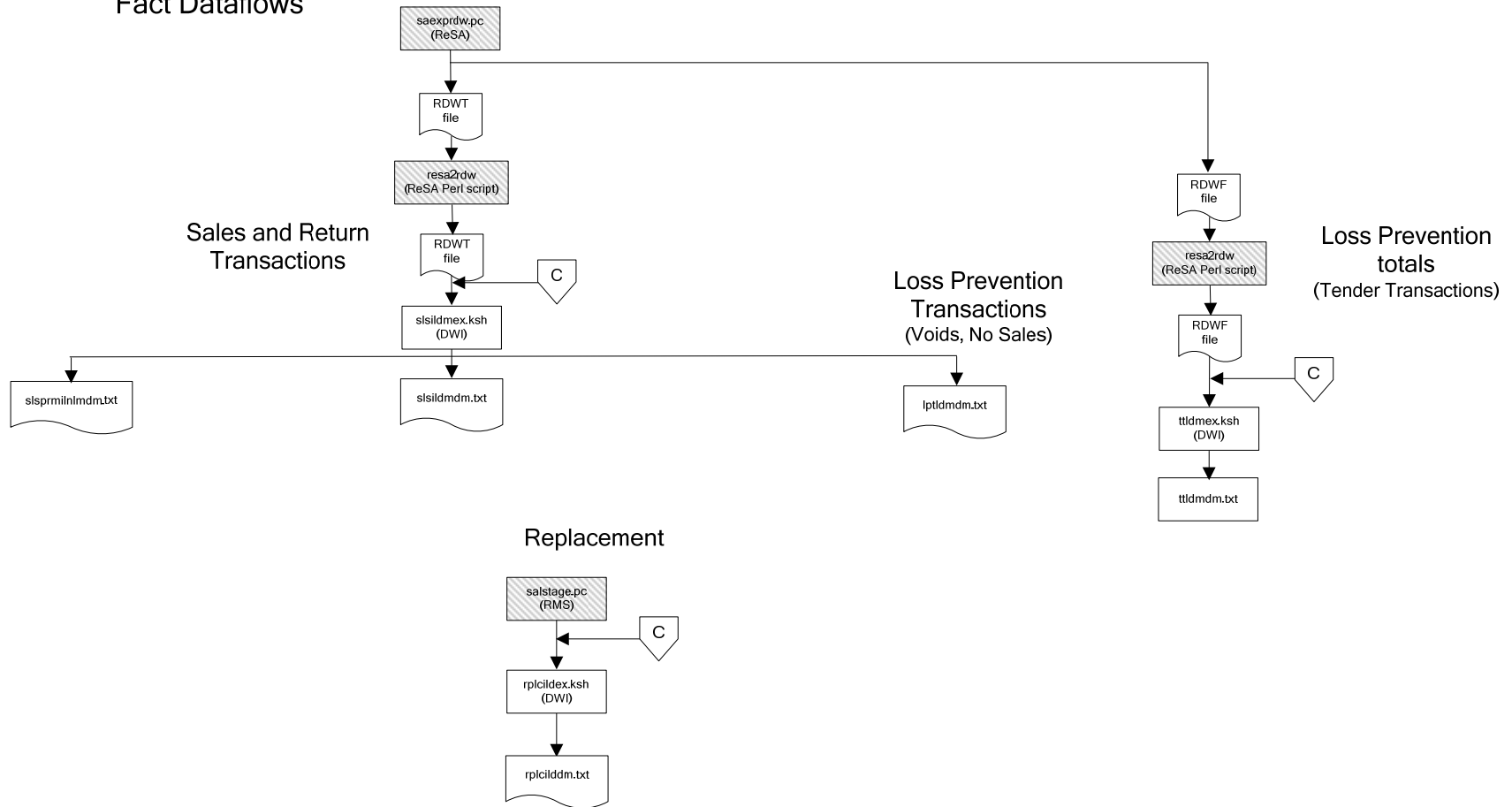


### Sales Forecasts

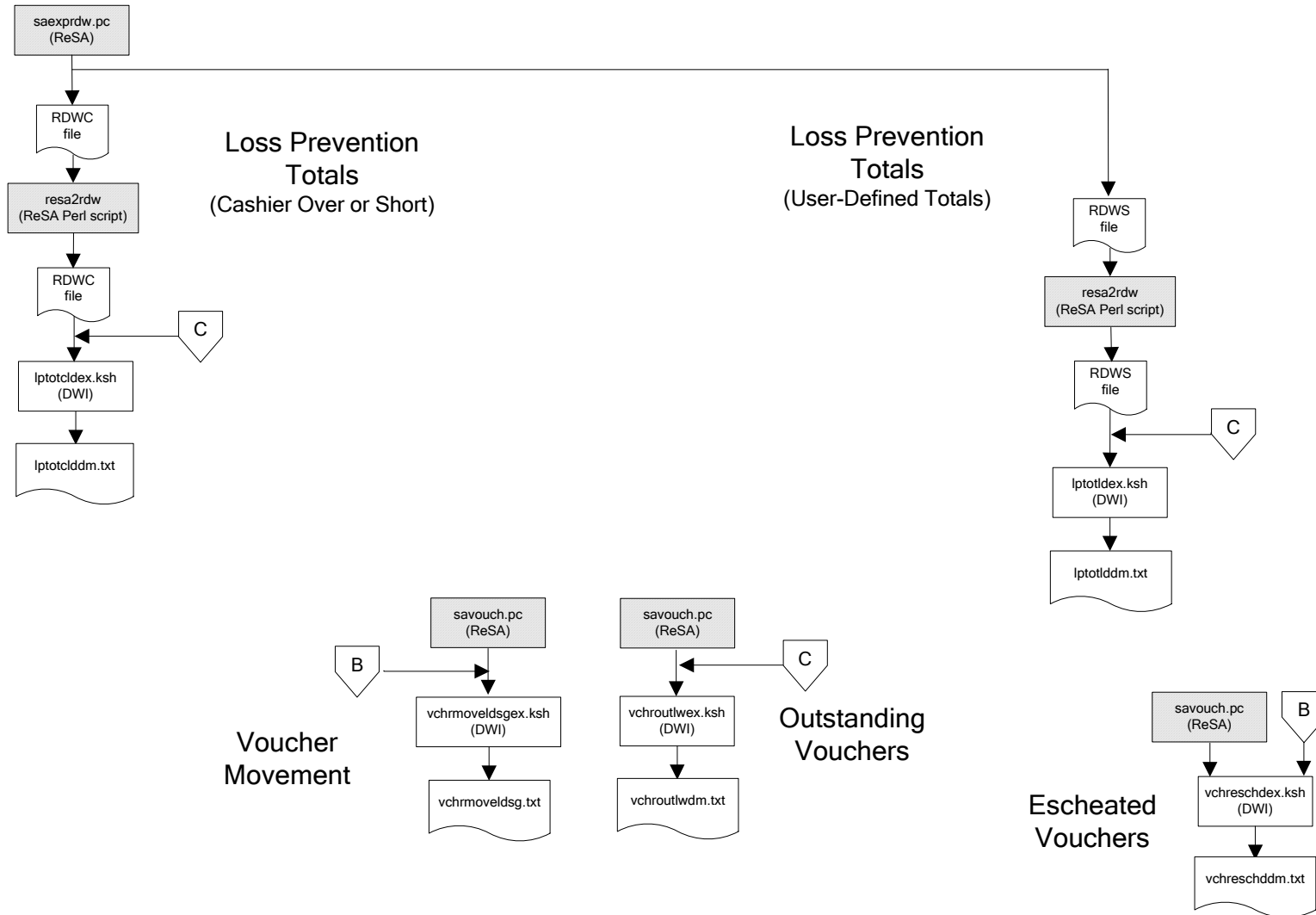


**Note:**  
Run sales forecast fact loads  
once weekly.

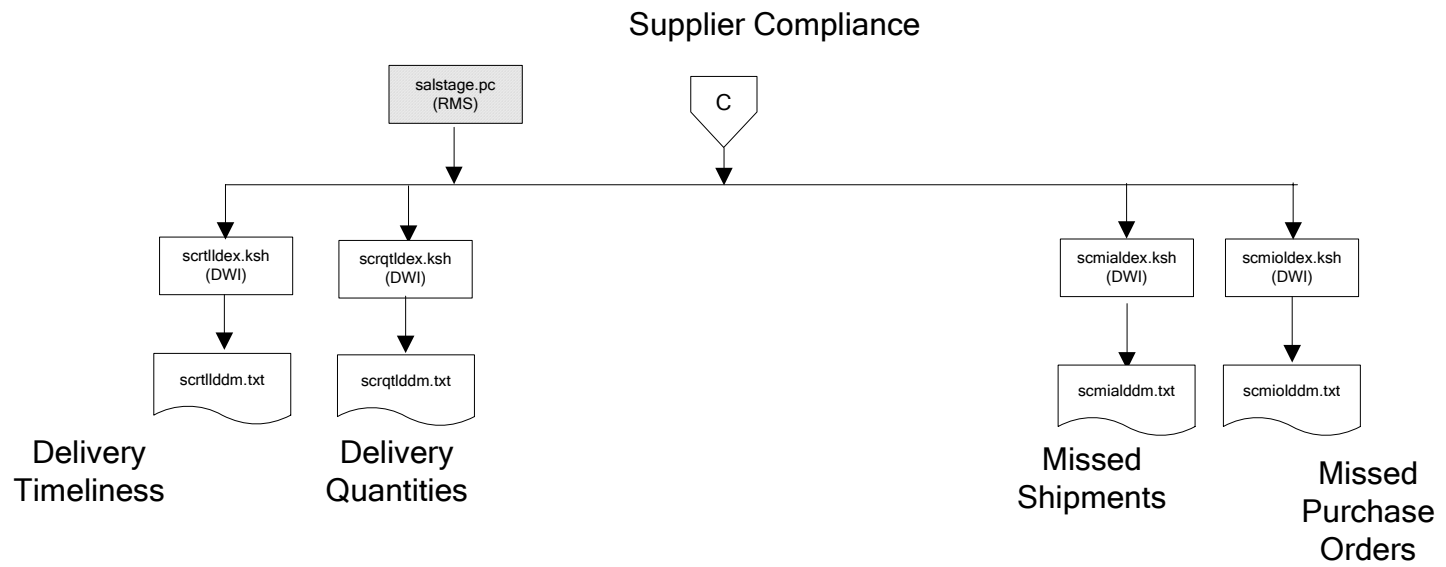
## Fact Dataflows



## Fact Dataflows



## Fact Dataflows



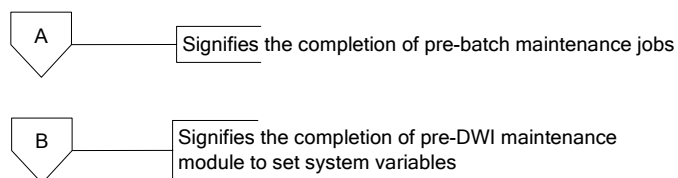
## Interface Diagram for RPM and RDW

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.

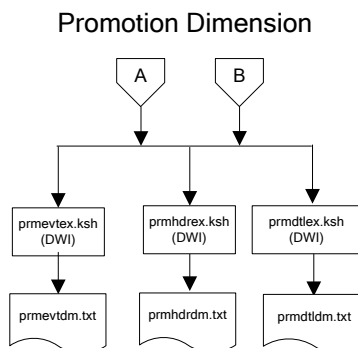
For detailed information about dimensions and facts, see the Retail Data Warehouse Operations Guide.

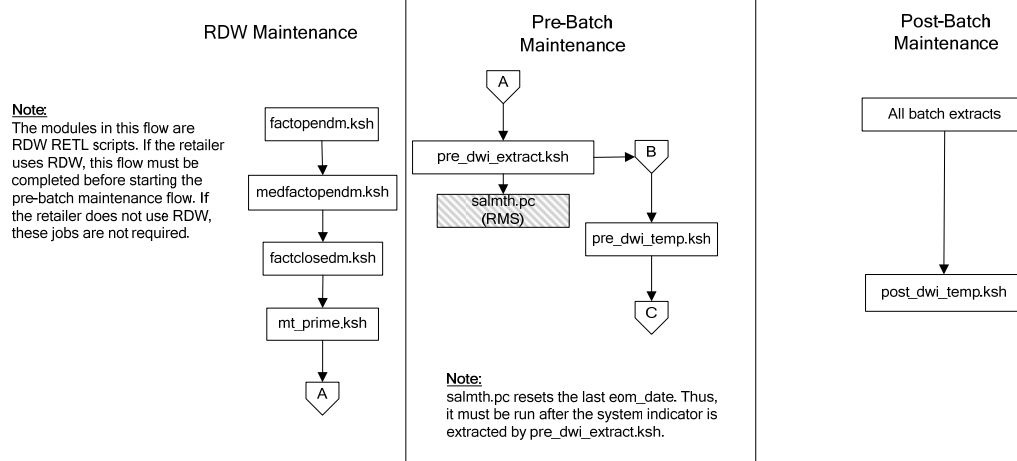
See the Retail Merchandising System Operations Guide Volume 1—Batch Overviews and Designs for more information about the modules shown in the following diagram.

### Legend



### Program Flow Diagram







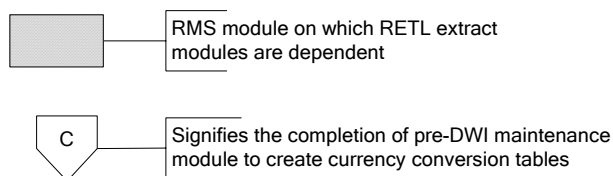
## Interface Diagram for ReIM and RDW

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 Retail Data Warehouse Operations Guide.

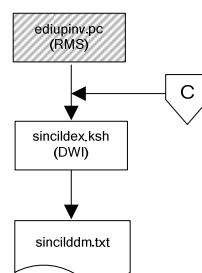
See the Retail Merchandising System Operations Guide Volume 1—Batch Overviews and Designs for more information about the modules shown in the following diagram.

### Legend



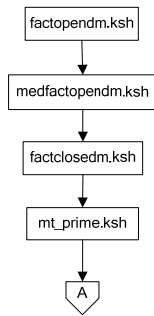
### Program Flow Diagram

#### Supplier Invoice Cost

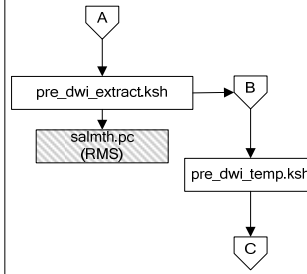


**Note:**  
The modules in this flow are RDW RETL scripts. If the retailer uses RDW, this flow must be completed before starting the pre-batch maintenance flow. If the retailer does not use RDW, these jobs are not required.

## RDW Maintenance



## Pre-Batch Maintenance



**Note:**  
salmth.pc resets the last eom\_date. Thus, it must be run after the system indicator is extracted by pre\_dwi\_extract.ksh.

## Post-Batch Maintenance

