



# **Logistics Management Configuration Guide**

Release 7.5 SP1

May 2006

---

# Copyright Notice

Logistics Management Configuration Guide, Release 7.5 SP1

Copyright © 2003 - 2006  
Yantra Corporation  
ALL RIGHTS RESERVED

**WARNING: ANY UNAUTHORIZED DUPLICATION OF THIS DOCUMENTATION SHALL BE AN INFRINGEMENT OF COPYRIGHT**

## Trade Secret Notice

This documentation, the software it describes, and the information and know-how they contain constitute the unpublished, proprietary, confidential and valuable trade secret information of Yantra Corporation, its affiliated companies or its or their licensors, and may not be used for any unauthorized purpose, or disclosed to others without the prior written permission of the applicable Yantra entity. This documentation and the software it describes have been provided pursuant to a license agreement that contains prohibitions against and/or restrictions on their copying, modification and use. Duplication, in whole or in part, if and when permitted, shall bear this notice and the Yantra Corporation copyright notice.

This documentation and the software it describes are licensed either "AS IS" or with a limited warranty, as set forth in the Yantra license agreement. Other than any limited warranties provided, NO OTHER WARRANTY IS EXPRESSED AND NONE SHALL BE IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE OR FOR A PARTICULAR PURPOSE. The applicable Yantra entity reserves the right to revise this documentation from time to time and to make changes in content hereof without the obligation to notify any person or entity of such revisions or changes.

Yantra does not warrant or represent that use of the software described in this documentation will insure compliance with the U.S. Department of Transportation's (DOT) Hazardous Materials Regulations (HMR) found in Title 49 of the Code of Federal Regulations (49 CFR), and users of this software should consult independent legal counsel and technical support to insure compliance with the HMR and other legal requirements.

**U.S. Government Restricted Rights.** This documentation and the software it describes are each and collectively a "commercial item" as defined in 48 C.F.R. 2.101. Consequently, as and when provided to any agency or instrumentality of the U.S. Government or to a U.S. Government prime contractor or a subcontractor at any tier ("Government Licensee"), the terms and conditions of the customary Yantra commercial license agreement are imposed on Government Licensees per 48 C.F.R. 12.212 or § 227.7201, as applicable, or through 48 C.F.R. § 52.244-6.

Yantra and the Yantra logo are marks of Yantra Corporation. All other services and products or company names are used for identification purposes only and may be marks of their respective owners.

---

Yantra Corporation  
One Park West  
Tewksbury, MA 01876  
1-978-513-6000

Copyright © 2003 -  
2006

# Contents

---

## Preface

Intended Audience .....	xi
Structure .....	xi
Yantra 7x Documentation.....	xiii
Conventions .....	xiv

## 1 Introduction

1.1 Business Models.....	2
1.1.1 Multi-Divisional Corporation .....	2
1.1.2 Third-Party Logistics .....	2
1.1.3 Marketplace.....	3
1.2 Logistics Management Configuration .....	3
1.2.1 Carrier Services .....	5
1.2.2 Carrier Special Services .....	5
1.2.3 Modification Reasons .....	5
1.2.4 Load Execution Process Model .....	5
1.2.5 Instruction Types .....	5
1.2.6 Charge Categories.....	6
1.2.7 Purge Criteria .....	6
1.2.8 Load Type .....	6
1.2.9 Stop Type .....	6
1.2.10 Modification Rules .....	6
1.2.11 Logistics Components .....	6

## 2 Navigating in the Configurator

2.1	Starting the Yantra 7x Configurator .....	7
2.2	The Yantra 7x Configurator Layout .....	8
2.2.1	Application Rules Side Panel .....	10
2.2.1.1	Accessing Configuration Screens.....	11
2.2.1.2	Determining Inheritance.....	11
2.2.1.3	Loading Another Organization's Rules.....	17
2.2.2	Work Area .....	19
2.2.2.1	Search Window .....	19
2.2.2.2	List Window .....	20
2.2.2.3	Details Window .....	21
2.2.2.4	Drag and Drop Window .....	22
2.3	Actions Available Throughout the Yantra 7x Configurator .....	24
2.3.1	Using Configurator's Lookup Functionality .....	24
2.3.2	Viewing the Document Types Associated with an Application .....	25
2.3.2.1	Adding a Document Type to an Application .....	26
2.3.3	Viewing the User Logged into the Configurator .....	27
2.3.4	Using Lists and List Filtering.....	27
2.3.5	Date and Time Entry .....	28
2.3.6	Using On-Line Help .....	29
2.3.7	Troubleshooting Errors .....	29
2.3.8	Using Special Characters .....	29

## 3 Configuring Cross Application Carrier Services

3.1	Creating a Carrier Service Code.....	31
3.2	Modifying a Carrier Service Code .....	33
3.3	Deleting a Carrier Service Code.....	33

## 4 Configuring Cross Application Carrier Special Services

4.1	Creating a Carrier Special Service Code.....	36
4.2	Modifying a Carrier Special Service Code .....	37
4.3	Deleting a Carrier Special Service Code .....	38

## **5 Configuring Transportation**

5.1	Creating an Activity Code .....	39
5.2	Modifying an Activity Code .....	40
5.3	Deleting an Activity Code .....	41

## **6 Configuring a Document's Modification Reasons**

6.1	Creating a Modification Reason .....	43
6.2	Modifying a Modification Reason .....	45
6.3	Deleting a Modification Reason .....	45

## **7 Configuring a Load Document's Pipeline**

7.1	Defining Process Type Details .....	47
7.2	Process Type Pipeline Configuration .....	48
7.2.1	Defining Pipeline Determination .....	48
7.2.1.1	Condition Variables for Pipeline Determination .....	49
7.2.2	Pipelines .....	49
7.2.3	Transactions .....	51
7.2.4	Statuses .....	53
7.2.5	Conditions .....	56
7.2.6	Actions .....	57

## **8 Configuring a Document's Instruction Types**

8.1	Creating an Instruction Type .....	59
8.2	Modifying an Instruction Type .....	60
8.3	Deleting an Instruction Type .....	60

## **9 Configuring a Load Document's Charge Categories**

9.1	Creating a Charge Category .....	63
9.1.1	Adding a Charge Name Associated with a Charge Category .....	65
9.1.2	Modifying a Charge Name Associated with a Charge Category .....	66
9.1.3	Deleting a Charge Name Associated with a Charge Category .....	66
9.2	Modifying a Charge Category .....	66
9.3	Deleting a Charge Category .....	67

<b>10</b>	<b>Configuring a Load Document's Purge Criteria</b>	
10.1	Modifying a Load Document Type's Purge Criteria Rule .....	69
<b>11</b>	<b>Configuring a Load Document's Load Types</b>	
11.1	Creating a Load Type .....	73
11.2	Modifying a Load Type .....	74
11.3	Deleting a Load Type .....	74
<b>12</b>	<b>Configuring a Load Document's Stop Types</b>	
12.1	Modifying a Stop Type .....	77
<b>13</b>	<b>Configuring a Load Document's Modification Rules</b>	
13.1	Changing a Load Document Type's Modification Rules .....	81
<b>14</b>	<b>Configuring Logistics Components</b>	
14.1	Defining Logistics .....	83
14.1.1	Defining Freight Terms .....	83
14.1.1.1	Creating a Freight Term .....	84
14.1.1.2	Modifying a Freight Term .....	86
14.1.1.3	Deleting a Freight Term .....	86
14.1.2	Defining Shipment Planning .....	86
14.1.3	Creating a Routing Guide .....	90
14.1.4	Modifying a Routing Guide .....	92
14.1.4.1	Creating a Routing Guide Line .....	92
14.1.4.2	Modifying a Routing Guide Line .....	100
14.1.4.3	Deleting a Routing Guide Line .....	101
14.1.5	Deleting a Routing Guide .....	101
<b>A</b>	<b>Time-Triggered Transaction Reference</b>	
A.1	Running Time-Triggered Transactions .....	104
A.2	Business Process Time-Triggered Transactions .....	105
A.2.1	Change Load Status .....	105
A.2.2	Change Shipment Status .....	107

A.2.3	Close Delivery Plan .....	108
A.2.4	Close Load .....	110
A.2.5	Close Manifest .....	112
A.2.6	Close Order .....	115
A.2.7	Close Receipts .....	117
A.2.8	Close Shipment.....	119
A.2.9	Collect Shipment Statistics .....	121
A.2.10	Complete Planned Order .....	123
A.2.11	Consolidate Additional Inventory .....	124
A.2.12	Consolidate To Shipment.....	126
A.2.13	Create Chained Order .....	129
A.2.14	Create Derived Order.....	131
A.2.15	Create Order Invoice .....	133
A.2.16	Create Shipment Invoice .....	135
A.2.17	ESP Evaluator .....	137
A.2.18	Mark Load as Trailer Loaded .....	139
A.2.19	Match Inventory.....	140
A.2.20	Payment Collection.....	142
A.2.21	Payment Execution.....	145
A.2.22	Post Inventory Match.....	148
A.2.23	Process Order Hold Type .....	150
A.2.24	Process Work Order Hold Type .....	152
A.2.25	Publish Negotiation Results .....	154
A.2.26	Release.....	156
A.2.27	Route Shipment .....	159
A.2.28	Schedule.....	161
A.2.29	Send Invoice .....	166
A.2.30	Send Order .....	168
A.2.31	Send Release.....	170
A.2.32	Start Order Negotiation.....	171
A.3	Time-Triggered Purge Transactions.....	173
A.3.1	Purge Strategy .....	174
A.3.2	Configuring Purge Transaction Log Files .....	174
A.3.3	Available Purges.....	175
A.3.3.1	Alert Purge.....	176

A.3.3.2	Capacity Purge.....	178
A.3.3.3	Delivery Plan Purge .....	181
A.3.3.4	Export Table Purge .....	183
A.3.3.5	Import Table Purge.....	185
A.3.3.6	Inventory Audit Purge .....	187
A.3.3.7	Inventory Purge .....	189
A.3.3.8	Inventory Supply Temp Purge .....	191
A.3.3.9	Load Purge .....	193
A.3.3.10	Manifest Purge .....	196
A.3.3.11	Negotiation History Purge .....	198
A.3.3.12	Negotiation Purge.....	200
A.3.3.13	Order History Purge .....	203
A.3.3.14	Order Purge.....	206
A.3.3.15	Order Status Audit Purge.....	211
A.3.3.16	Picklist Purge .....	213
A.3.3.17	Price List Purge .....	215
A.3.3.18	Receipt History Purge.....	217
A.3.3.19	Receipt Purge .....	219
A.3.3.20	Reprocess Error Purge.....	222
A.3.3.21	Reservation Purge .....	224
A.3.3.22	Shipment History Purge.....	225
A.3.3.23	Shipment Purge .....	228
A.3.3.24	Shipment Statistics Purge.....	231
A.3.3.25	Statistics Purge .....	233
A.3.3.26	Work Order History Purge.....	235
A.3.3.27	Work Order Purge .....	237
A.4	Task Queue Syncher Time-Triggered Transactions .....	240
A.4.1	Load Execution Task Queue Syncher .....	241
A.4.2	Order Delivery Task Queue Syncher.....	242
A.4.3	Order Fulfillment Task Queue Syncher.....	243
A.4.4	Order Negotiation Task Queue Syncher .....	244
A.5	Monitors .....	246
A.5.1	Availability Monitor .....	246
A.5.2	Exception Monitor.....	248
A.5.3	Inventory Monitor.....	250



A.5.4	Negotiation Monitor .....	251
A.5.5	Order Monitor .....	253
A.5.6	Enhanced Order Monitor.....	255
A.5.7	Enhanced Return Monitor .....	258
A.5.8	Real-time Availability Monitor.....	262
A.5.9	Shipment Monitor.....	266
A.5.10	Work Order Monitor .....	268

## **B Condition Builder Attributes**

B.1	Sales Order .....	273
B.1.1	Order Fulfillment .....	273
B.1.2	Order Negotiation.....	275
B.1.3	Outbound Shipment .....	276
B.2	Planned Order .....	277
B.2.1	Planned Order Execution .....	277
B.2.2	Planned Order Negotiation.....	278
B.3	Return Order .....	278
B.3.1	Reverse Logistics .....	278
B.3.2	Return Shipment.....	280
B.3.3	Return Receipt.....	280
B.4	Template Order.....	281
B.5	Purchase Order .....	281
B.5.1	Purchase Order Execution.....	281
B.5.2	Purchase Order Negotiation .....	284
B.5.3	Inbound Shipment.....	284
B.5.4	Purchase Order Receipt .....	284
B.6	Transfer Order .....	285
B.6.1	Transfer Order Execution.....	285
B.6.2	Transfer Order Delivery.....	285
B.6.3	Transfer Order Receipt.....	285
B.7	Load Execution .....	285
B.8	General .....	286
B.9	WMS Putaway.....	287
B.10	WMS Layout Definition.....	287
B.11	WMS Inventory.....	287

B.12	Trailer Loading.....	287
B.13	Task Execution.....	288
B.14	Move Request Execution.....	288
B.15	Manifesting.....	288
B.16	Over Pack Build.....	288
B.17	Count Execution.....	288
B.18	Pack Process.....	290
B.19	Outbound Picking .....	292
B.20	VAS Process .....	293

## Index

# Preface

---

This manual describes how to use the Yantra 7x Configurator.

## Intended Audience

This manual is intended for use by system administrators and managers who need to configure Yantra 7x rules and business processes as they pertain to their logistics management business practices.

## Structure

This manual contains the following sections:

### **Chapter 1, "Introduction"**

This chapter briefly describes the contents of this guide.

### **Chapter 2, "Navigating in the Configurator"**

This chapter explains the layout of the Yantra 7x Configurator, actions you can perform throughout the application, and important concepts you should be aware of before using the application.

### **Chapter 3, "Configuring Cross Application Carrier Services"**

This chapter explains how you can configure codes to identify the different carrier services a Carrier uses to ship orders.

### **Chapter 4, "Configuring Cross Application Carrier Special Services"**

This chapter explains how you can configure codes to identify the different carrier special services a Carrier uses to ship and deliver orders.

### **Chapter 6, "Configuring a Document's Modification Reasons"**

This chapter explains how you can define common codes for modification reasons.

### **Chapter 7, "Configuring a Load Document's Pipeline"**

This chapter explains how you can configure a load document's pipeline using process types.

### **Chapter 8, "Configuring a Document's Instruction Types"**

This chapter explains how you can define the common codes used when adding special instructions to an order document.

### **Chapter 9, "Configuring a Load Document's Charge Categories"**

This chapter explains how you can define charge definitions that you can associate with orders and invoices through charge categories. These categories contain a group of related charge names that can be used when the particular category is used.

### **Chapter 10, "Configuring a Load Document's Purge Criteria"**

This chapter explains the purge criteria business rules that are used to define qualifications around each type of purge. Purges are the process by which old data is removed from the system database.

### **Chapter 11, "Configuring a Load Document's Load Types"**

This chapter explains how you can define codes for load types that appear on a load document.

### **Chapter 12, "Configuring a Load Document's Stop Types"**

This chapter explains how you can modify codes for stop types that appear on a load document.

### **Chapter 13, "Configuring a Load Document's Modification Rules"**

This chapter explains how you can configure a load document's modification rules.

### **Chapter 14, "Configuring Logistics Components"**

This chapter explains the configuration of components used by different logistics related functionality throughout the business application module.

### **Appendix A, "Time-Triggered Transaction Reference"**

This chapter explains time-triggered transactions that are utilities that perform a variety of individual functions, automatically and at specific time intervals.

## **Appendix B, "Condition Builder Attributes"**

This chapter explains the attributes used in the condition builder to build statements for each process type.

# **Yantra 7x Documentation**

For more information about the Yantra® 7x components, see the following manuals in the Yantra® 7x documentation set:

- *Yantra® 7x Release Notes*
- *Yantra® 7x Installation Guide*
- *Yantra® 7x Upgrade Guide*
- *Yantra® 7x Performance Management Guide*
- *Yantra® 7x High Availability Guide*
- *Yantra® 7x System Management Guide*
- *Yantra® 7x Localization Guide*
- *Yantra® 7x Customization Guide*
- *Yantra® 7x Integration Guide*
- *Yantra® 7x Product Concepts*
- *Yantra® 7x Warehouse Management System Concepts Guide*
- *Yantra® 7x Platform Configuration Guide*
- *Yantra® 7x Distributed Order Management Configuration Guide*
- *Yantra® 7x Supply Collaboration Configuration Guide*
- *Yantra® 7x Inventory Synchronization Configuration Guide*
- *Yantra® 7x Product Management Configuration Guide*
- *Yantra® 7x Logistics Management Configuration Guide*
- *Yantra® 7x Reverse Logistics Configuration Guide*
- *Yantra® 7x Warehouse Management System Configuration Guide*

- *Yantra® 7x Platform User Guide*
- *Yantra® 7x Distributed Order Management User Guide*
- *Yantra® 7x Supply Collaboration User Guide*
- *Yantra® 7x Inventory Synchronization User Guide*
- *Yantra® 7x Logistics Management User Guide*
- *Yantra® 7x Reverse Logistics User Guide*
- *Yantra® 7x Warehouse Management System User Guide*
- *Yantra® 7x Mobile Application User Guide*
- *Yantra® 7x Analytics Guide*
- *Yantra® 7x Javadocs*
- *Yantra® 7x Glossary*
- *Yantra® 7x Carrier Server Guide*
- *Yantra® 7x Application Server Installation Guide* (for optional component)

## Conventions

The following conventions may be used in this manual:

Convention	Meaning
. . .	An ellipsis represents information that has been omitted.
< >	Angle brackets indicate user-supplied input.
mono-spaced text	Mono-spaced text indicates a file name, an API name, or a code example.
/ or \	Slashes and backslashes are file separators for Windows, UNIX and LINUX operating systems. The file separator for the Windows operating system is "\" and the file separator for Unix and Linux systems is "/". The Unix convention is used unless otherwise mentioned.

# Introduction

---

This book concentrates on the rules and setup configurations that make up the Logistics Management business application in the Yantra 7x Configurator. This book is intended for both Hub and Enterprise administrators using the Yantra 7x Configurator to set up the Yantra 7x environment. Business analysts should also use this book to plan appropriate business practices as they pertain to Yantra 7x. Programmers should refer to the *Yantra 7x Customization Guide* for information about extending Yantra 7x. System Integrators should refer to the *Yantra 7x Integration Guide* for information about integrating external applications with Yantra 7x.

**Important:** This book assumes that you have read and are familiar with the concepts and business functionality detailed in the *Yantra 7x Product Concepts*.

The Yantra 7x Configurator is a collection of all the rules and setup configurations necessary to implement Yantra 7x organized so that configuration can be done for each business application separately. The following business applications can be configured within the Yantra 7x Configurator:

- Yantra 7x Distributed Order Management
- Yantra 7x Inventory Synchronization
- Yantra 7x Product Management
- Yantra 7x Logistics Management
- Yantra 7x Supply Collaboration
- Yantra 7x Reverse Logistics

- Yantra 7x Warehouse Management System
- Yantra 7x Platform

# 1.1 Business Models

There is no single business model that encompasses the environment in which all the Yantra 7x applications can be used. Therefore, there is no single way to configure your Yantra 7x environment.

For example, your company might be considered a multi-divisional corporation, a third-party logistics company, or a marketplace business. Each of these business models require a different conceptual approach to Yantra 7x configuration.

## 1.1.1 Multi-Divisional Corporation

The **multi-divisional corporation model** is a business corporation whose primary focus is managing purchase and sales activities. A typical multi-divisional corporation can be a buyer, a seller, or both. It could also be a retailer, a manufacturer, or both. Whatever form the multi-divisional corporation takes, it normally has multiple channels with different types of customers, such as, consumers, retailers, dealers, and original equipment manufacturers.

In the multi-divisional corporation model, each division might be set up as an Enterprise in Yantra 7x. This setup allows both segregation of transactions by division and global visibility at the corporate level. Each Enterprise configures their own business rules, workflow, and transaction processing.

## 1.1.2 Third-Party Logistics

Traditional **third-party logistics** companies provide a range of outsourced services such as warehousing, transportation, and contract manufacturing.

Large companies can gain the competitive advantage through the real-time management of their supply chains. These advantages include lower costs and improved customer service. Additionally, new sales channels such as web stores, hand-held devices, and in-store kiosks provide companies new methods of reaching their customers. All of these issues have increased the complexity of the fulfillment process.



Yantra 7x provides the engine needed to run the operations of a contract fulfillment provider as well as a centralized system for real-time order execution and event driven problem solving for an entire fulfillment network. It enables fulfillment providers to configure the fulfillment process to meet the needs of their clients.

In the third-party logistics model, each client might be set up as an Enterprise. This setup allows the third-party logistics Hub to have visibility of all transactions in the Hub environment, while the clients that are set up as Enterprises only have visibility to their own transactions. This allows the third-party logistics business to provide unique transaction processing to its clients.

### 1.1.3 Marketplace

A **marketplace** is an online intermediary that connects Buyers and Sellers. Marketplaces eliminate inefficiencies by aggregating offerings from many Sellers or by matching Buyers and Sellers in an exchange or auction. For Buyers, they lower purchasing costs and help them reach new Sellers. For Sellers, they lower sales costs and give them access to new customers. It is a central location, or Hub, where a trusted intermediary integrates both procedures and technology to lower the costs and enhance the effectiveness of Buyer and Seller transactions.

In the marketplace model, each market might be set up as an Enterprise. This setup allows each market to be unique with their own product or service handling.

## 1.2 Logistics Management Configuration

The Yantra 7x Logistics Management application is a collection of common components used for creating delivery plans in the Yantra 7x Application Consoles.

A delivery plan is a complete sequence of movements needed to deliver one or more orders from one or multiple origins to one or multiple destinations. A delivery plan is comprised of shipments, loads, origins, stops, and destinations.

### Shipment

A shipment is a delivery of one or more orders and order lines from a single shipper to a single consignee. A shipment can be carried through multiple loads and by multiple carriers.

### Load

A load carries one or more complete shipments (never a partial shipment) between two points. A load has one origin and one destination, but it can have multiple intermediate stops. Shipments can be added to a load at its origin or any intermediate stop and can be dropped off at the load destination or any intermediate stop.

### Origin

An origin is the node the load originally ships from.

### Stop

A stop is any location where a shipment is picked up or dropped off. A load has a stop sequence that determines its travel route.

### Destination

A destination is the last node or address in the load's travel route where all remaining shipments in a load are dropped off.

In the Yantra 7x Configurator you can use the Logistics Management configuration grouping to establish the following aspects of Yantra 7x for your business applications:

- [Carrier Services](#)
- [Carrier Special Services](#)
- [Modification Reasons](#)
- [Instruction Types](#)
- [Charge Categories](#)
- [Purge Criteria](#)
- [Load Type](#)
- [Stop Type](#)
- [Modification Rules](#)

- [Logistics Components](#)

### 1.2.1 Carrier Services

Carrier Services is used for defining codes that identify the different carrier services a Carrier can use to ship orders. For more information about Carrier Services, see [Chapter 4, "Configuring Cross Application Carrier Special Services"](#).

### 1.2.2 Carrier Special Services

Carrier Special Services is used for defining codes that identify the different carrier special services a Carrier can use to ship and deliver orders. For more information about Carrier Special Services, see [Chapter 4, "Configuring Cross Application Carrier Special Services"](#).

### 1.2.3 Modification Reasons

Modification Reasons is used to define common codes for **modification reasons**. These codes define why a modification was made by a user. For more information about Modification Reasons, see [Chapter 6, "Configuring a Document's Modification Reasons"](#).

### 1.2.4 Load Execution Process Model

You can define a load document's business process workflow by creating process type pipelines. A **process type pipeline** is a series of transactions and statuses that guide a load document through a predefined process. A pipeline consists of the different statuses a document goes through during a load's planning. You can also set up transactions consisting of events, actions, and conditions, as they pertain to the pipeline you are configuring. For more information about Load Execution Process Model, see [Chapter 7, "Configuring a Load Document's Pipeline"](#).

### 1.2.5 Instruction Types

Instruction Types is used to define the common codes used when adding special instructions to a load document. For more information about Instruction Types, see [Chapter 8, "Configuring a Document's Instruction Types"](#).

### 1.2.6 Charge Categories

Charge Categories is used to define **charge definitions** that you can associate with orders and invoices through charge categories. These categories contain a group of related charge names that can be used when the particular category is used. For more information about Charge Categories, see [Chapter 9, "Configuring a Load Document's Charge Categories"](#).

### 1.2.7 Purge Criteria

Purge Criteria is used to define the parameters used when purging load document related records from the system. For more information about Purge Criteria, see [Chapter 10, "Configuring a Load Document's Purge Criteria"](#).

### 1.2.8 Load Type

Load Type is used to define codes for load types that appear on a load document. For more information about Load Type, see [Chapter 11, "Configuring a Load Document's Load Types"](#).

### 1.2.9 Stop Type

Stop Type is used to modify codes for stop types that appear on a load document. For more information about Stop Types, see [Chapter 12, "Configuring a Load Document's Stop Types"](#).

### 1.2.10 Modification Rules

Modification Rules is used to define rules that determine when modifications can be made during a load document's life cycle. For more information about Modification Rules, see [Chapter 13, "Configuring a Load Document's Modification Rules"](#).

### 1.2.11 Logistics Components

Logistics Components is used to define the different logistics related functionality throughout the business application module. For more information about Logistics Components, see [Chapter 14, "Configuring Logistics Components"](#).

## Navigating in the Configurator

---

This chapter discusses the layout of the Yantra 7x Configurator, actions you can perform throughout the application, and important concepts you should be aware of before using the application.

### 2.1 Starting the Yantra 7x Configurator

To access the Yantra 7x Configurator:

1. Point your browser to `http://<Yantra 7x installation server>/yantra/console/start.jsp`.

The browser displays the Sign In window.

2. Enter your login ID and password and choose the Sign In button. The Yantra 7x Application Consoles Home Page is displayed.
3. From the menu bar, choose Configuration > Launch Configurator. The Yantra 7x Configurator opens in a new window.

**Note:** Additionally, enterprise users who maintain an enterprise can access the Yantra 7x Configurator by means of `http://<Yantra 7x installation server>/yantra/console/login.jsp`.

**Note:** If both the Yantra 7x Configurator and the Yantra 7x System Management are opened at the same time, and if a dialogue window is opened in either application, the other will stop responding to user input until that dialogue window is closed. This is due to a bug in the Java platform.

## 2.2 The Yantra 7x Configurator Layout

The Yantra 7x Configurator is a graphical user interface that can be used to configure different aspects of Yantra 7x. The different configurations are defined by logical groupings called applications that can be accessed from the Configurator's menu bar.

*Figure 2–1 Applications Menu*



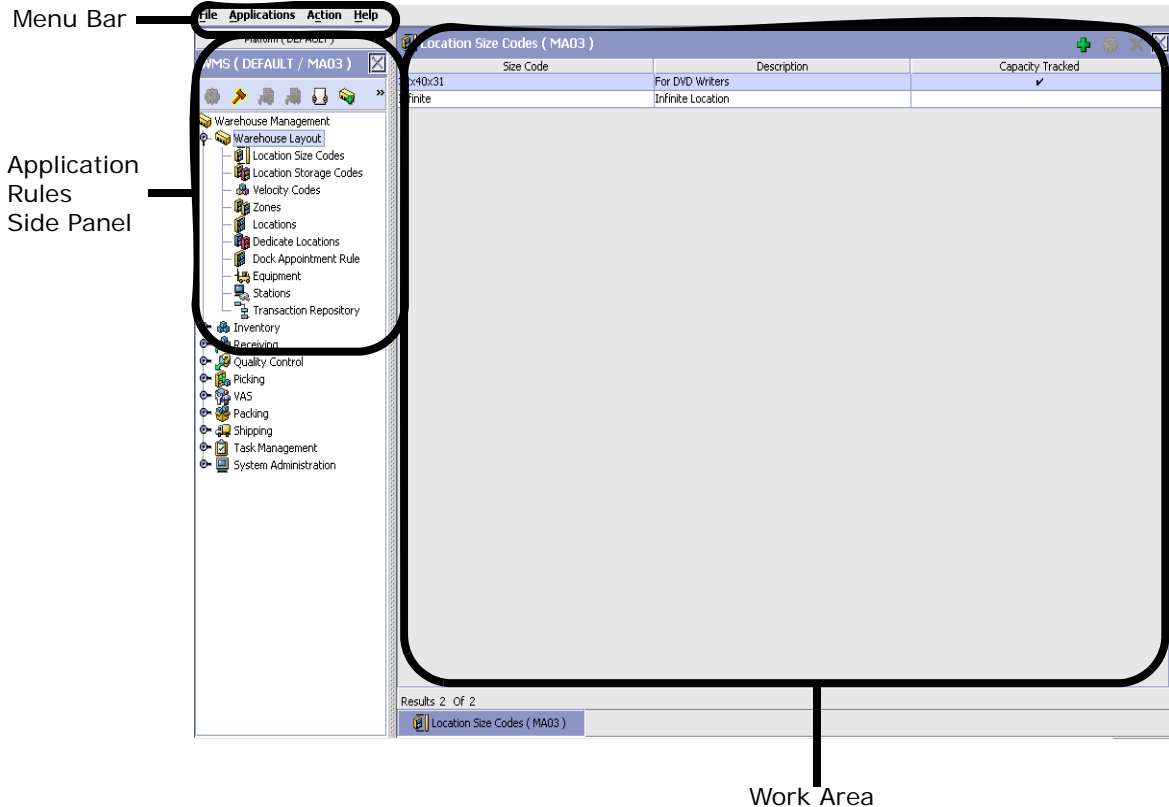
Each application focuses on a particular aspect of Yantra 7x and contains all of the rules, common codes, and settings necessary for Yantra 7x to work in a real-world business setting.

The following applications can be configured in this version of Yantra 7x:

- Distributed Order Management
- Inventory Synchronization
- Product Management
- Logistics Management
- Supply Collaboration
- Reverse Logistics
- Warehouse Management
- Platform

When you select the application that you want to configure, the Configurator displays a side panel containing all of the available configuration rules for the selected application and a work area in which these rules can be configured.

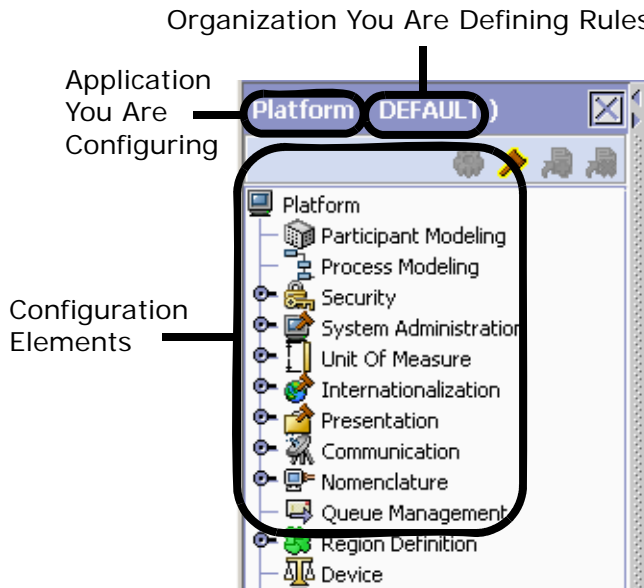
**Figure 2–2 The Standard Configurator Application Interface**



### 2.2.1 Application Rules Side Panel

The application rules side panel displays a hierarchical tree of elements specific to processes used within the application.



**Figure 2–3 Example of Application Rules Side Panel**

The application rules side panel also identifies the organization you are configuring rules for and what, if any, rules are inherited from another organization.

You can use the application rules side panel for:

- [Accessing Configuration Screens](#)
- [Determining Inheritance](#)
- [Loading Another Organization's Rules](#)

### 2.2.1.1 Accessing Configuration Screens

The main purpose of the application rules side panel is to provide an interface to access the application's individual configuration screens. To access a configuration screen, browse through the application tree and double-click on the applicable configuration element, the element's configuration screen is then displayed in the work area.

### 2.2.1.2 Determining Inheritance

In Yantra 7x, when an Enterprise is created it can inherit all or part of an existing Enterprise's configuration rules. This inheritance is done at the

configuration group level. A configuration group is a classification of similar configuration elements. For example, all of the rules and configurations dealing with items are grouped together into one configuration group and all of the rules and configurations dealing with organizations are grouped into another.

An administrator organization is set for every organization defined within the system. Only the administrator organization can modify the rules defined for a particular organization. If a particular organization administers multiple organizations, then they can load the rules of organization that it administers within the application tree. For more information about loading another organization's rules, see [Section 2.2.1.3, "Loading Another Organization's Rules"](#) on page 17.

Configuration groups are associated with organization levels. Organization levels determine how configuration groups are inherited and which organizations can maintain them. The organization levels defined in Yantra 7x are:

- Hub Level - Configuration groups that are associated with the Hub organization
- Enterprise Level - Configuration groups that are associated with the individual Enterprise organizations within the Hub environment
- Catalog Organization - Configuration groups that are associated with the organization(s) that maintains the catalog(s) within the Hub environment
- Inventory Organization - Configuration groups that are associated with the organization(s) that maintains the inventory within the Hub environment
- Organization - Configuration groups that are associated with any organization within the Hub environment.

The following table details the rules used to determine which organizations can maintain a configuration group as defined by the organization level. The table also describes the rules that determine how configuration groups are inherited when an organization is created.

**Table 2–1 Organization Level Rules**

Organization Level	Organizations That Can Modify at this Level...	Inheritance Details
Hub Level	Only the Hub organization can modify configuration groups at the Hub level. All other organizations have read-only access.	All organizations share this information.
Enterprise Level	Only Enterprise organizations can modify configuration groups at the Enterprise level.  Any business transaction requiring Enterprise configuration is picked up from the Enterprise established by the transactional context. For example, order documents have a specific Enterprise.	An Enterprise can inherit this configuration from another Enterprise. Additionally, this configuration can be overridden at a configuration group level.  When an Enterprise is created, it inherits Enterprise level rules from its primary Enterprise.
Catalog Organization	Organizations that are designated as catalog organizations can modify configuration groups at the catalog organization level.	None.
Inventory Organization	Organizations that are designated as inventory organizations can modify configuration groups at the inventory organization level.	None.
Organization	Any organization assigned a role (Seller, Buyer, etc.) can modify configuration groups at the organization level.	None.

---



---

**Important:** You cannot inherit from an Enterprise that does not have the same inventory, capacity, and catalog organizations as the organization you are configuring.

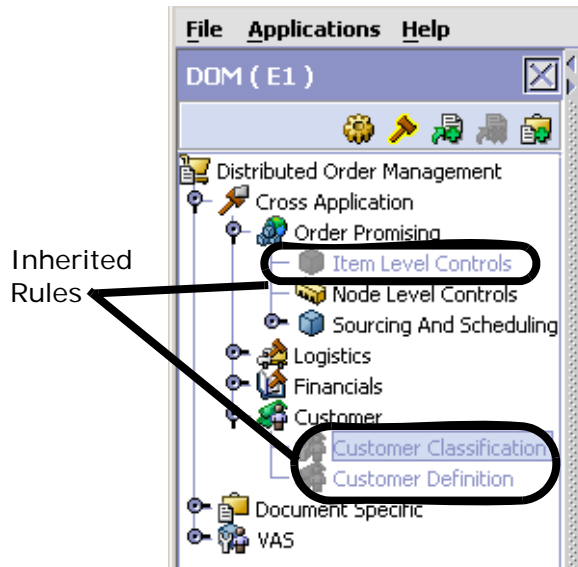
---



---

The application rules side panel displays rules that have been inherited as grayed out.

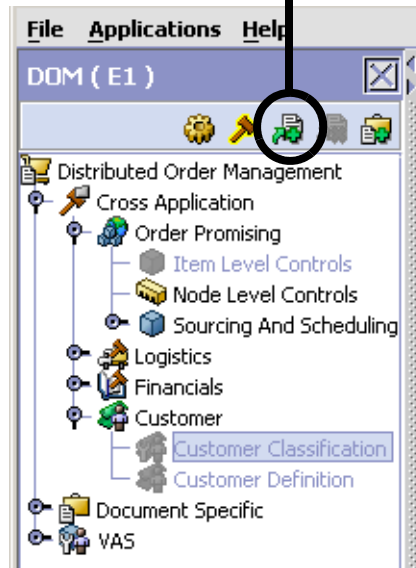
**Figure 2–4** *Inherited Rules in the Application Rules Side Panel*



As stated in the table above, depending on the organization you are logged in as, you may be able to override some inherited rules. If a rule can be overridden, the Override Configuration icon becomes available in the application rule side panel when you highlight the rule.

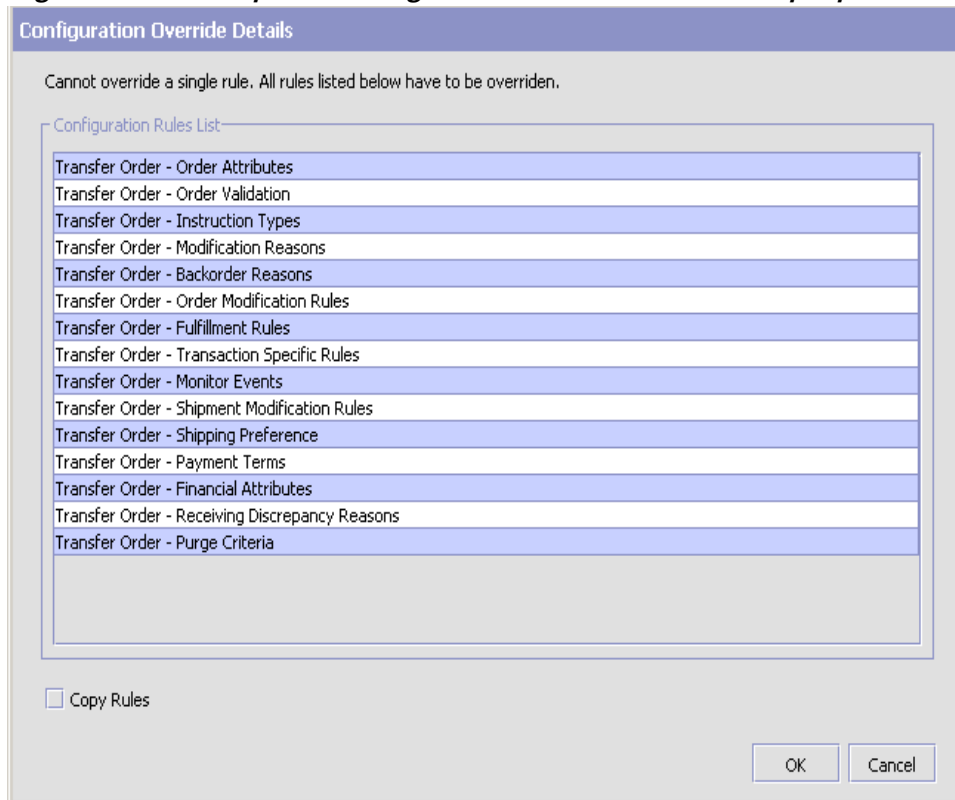
**Figure 2–5 Override Configuration Icon**

Override Configuration Icon is Available



When you choose to override a rule you also override any other rules in the configuration group the rule you are overriding is associated with. When you choose the Override Configuration icon the Configuration Override Details pop-up window is displayed. This window provides the list of rules that will be overridden.

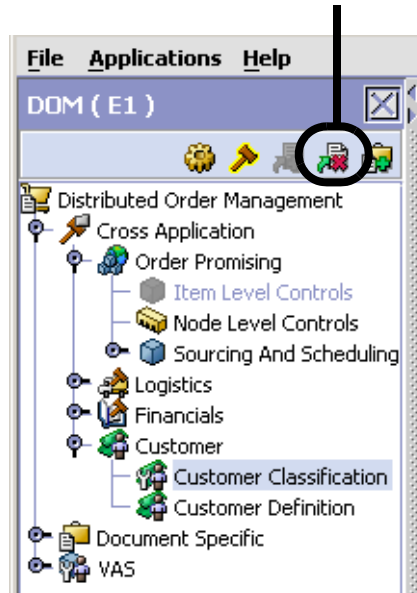
**Figure 2–6 Example of Configuration Override Details Pop-Up Window**



If you override a configuration group and then decide to "re-inherit" the original rules, you can choose the Give Back Configuration Ownership icon. This icon becomes available in the application rules side panel for rules that have been overridden.

**Figure 2–7 Give Back Configuration Ownership Icon**

Give Back Configuration Ownership Icon is Available



When you select the Give Back Configuration Ownership Icon, the Configuration Override Details pop-up window is displayed. This window provides the list of rules that will be re-inherited.

**Important:** If you select the Delete Rules field on the Configuration Override Details pop-up window, you give back rule ownership to the organization you originally inherited from, however you do not retain any of the rules that you inherited from them.

If you do not select this field, you give back rule ownership to the organization you originally inherited from, but you retain the rules that you inherited from them.

### 2.2.1.3 Loading Another Organization's Rules

An administrator organization is set for every organization defined within the system. Only the administrator organization can modify the rules defined for a particular organization. If a particular organization administers multiple organizations, then they can load the rules of


organization that it administers within the application tree. See [Table 2–1](#) for the rules that determine which organizations you can administer.

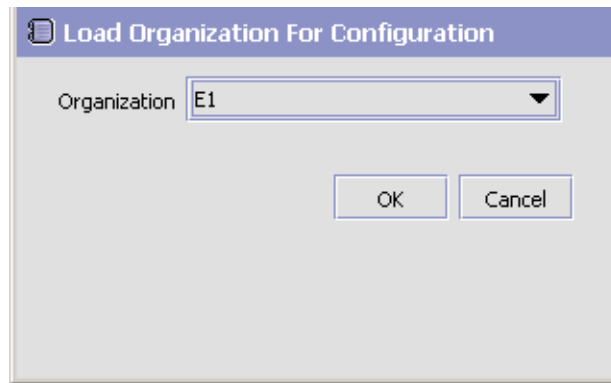
---

**Note:** The rules that are available from the tree in the application rules side panel may vary depending on the type of organization you select and the roles it has been assigned.

---

To load another organization's rules:

1. From the applicable application rules side panel, choose . The Load Organizations for Configuration pop-up window appears.



2. From Organization, select the organization that you want to work with.
3. Choose OK. The organization's rules are displayed in the application rules side panel.



---

---

**Note:** The application rules side panel displays the organization you are working with in parentheses.

---

---

## 2.2.2 Work Area

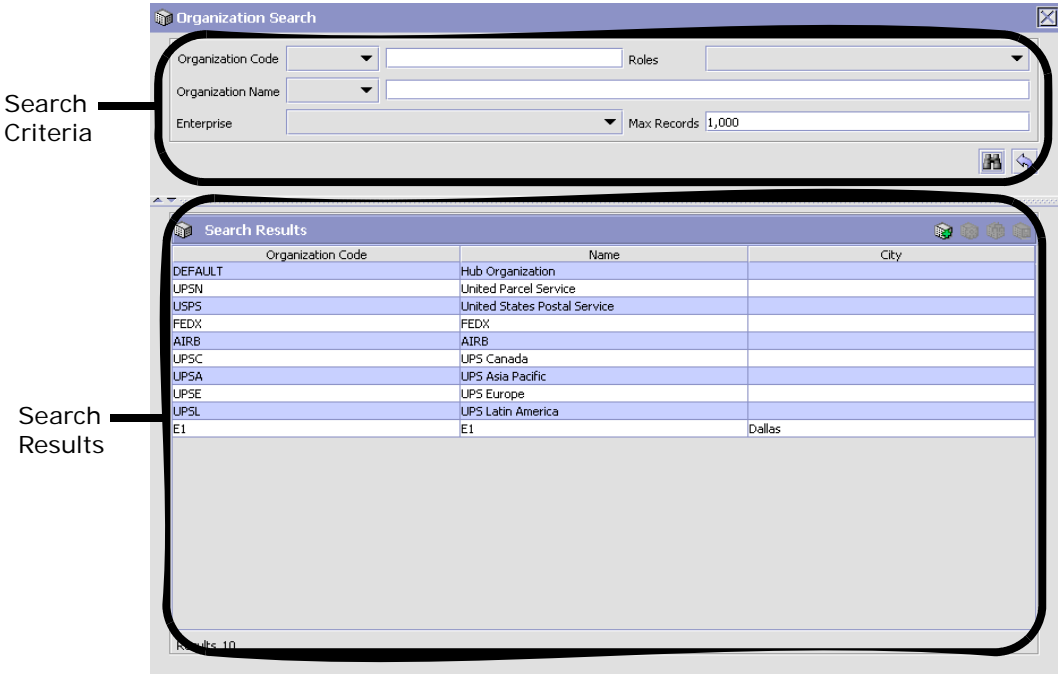
The work area is the main area in which different configuration screens appear. The following are the main types of screens that you will come across:

- [Search Window](#)
- [List Window](#)
- [Details Window](#)
- [Drag and Drop Window](#)

### 2.2.2.1 Search Window

A search window provides you with a means to perform a filtered search. The upper panel of a search window offers criteria applicable to the entity you are searching through which you can narrow your search. The lower panel lists the results of a search once it has been performed.

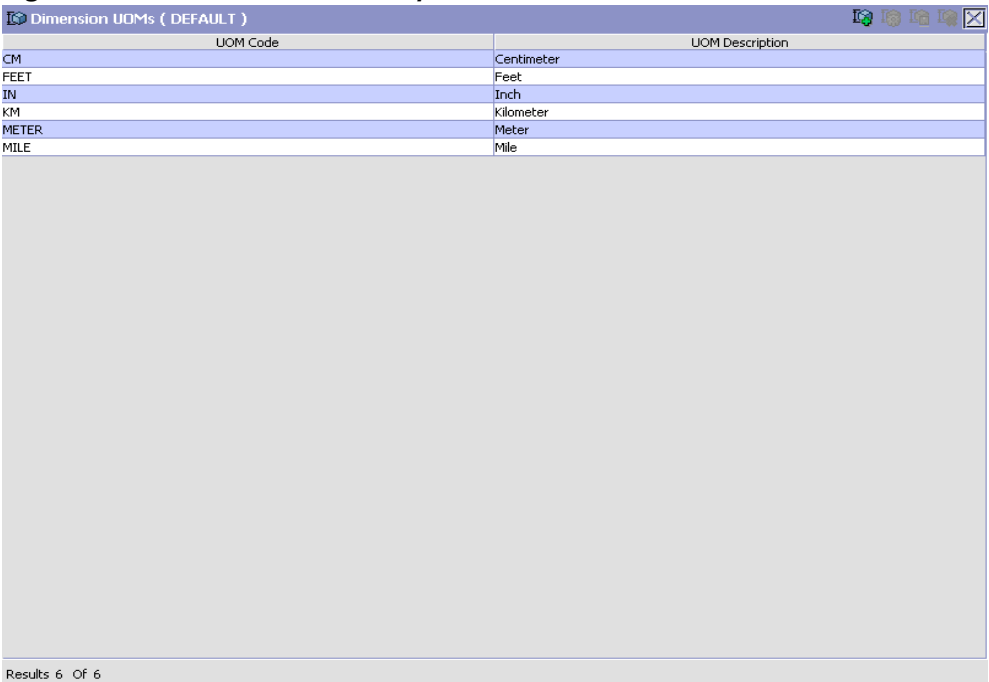
Figure 2–8 Search Window Example



### 2.2.2.2 List Window

When you choose to configure a specific rule or code that does not require a search, the Configurator may display a basic list window of the rules and codes that have previously been configured.

Figure 2–9 List Window Example



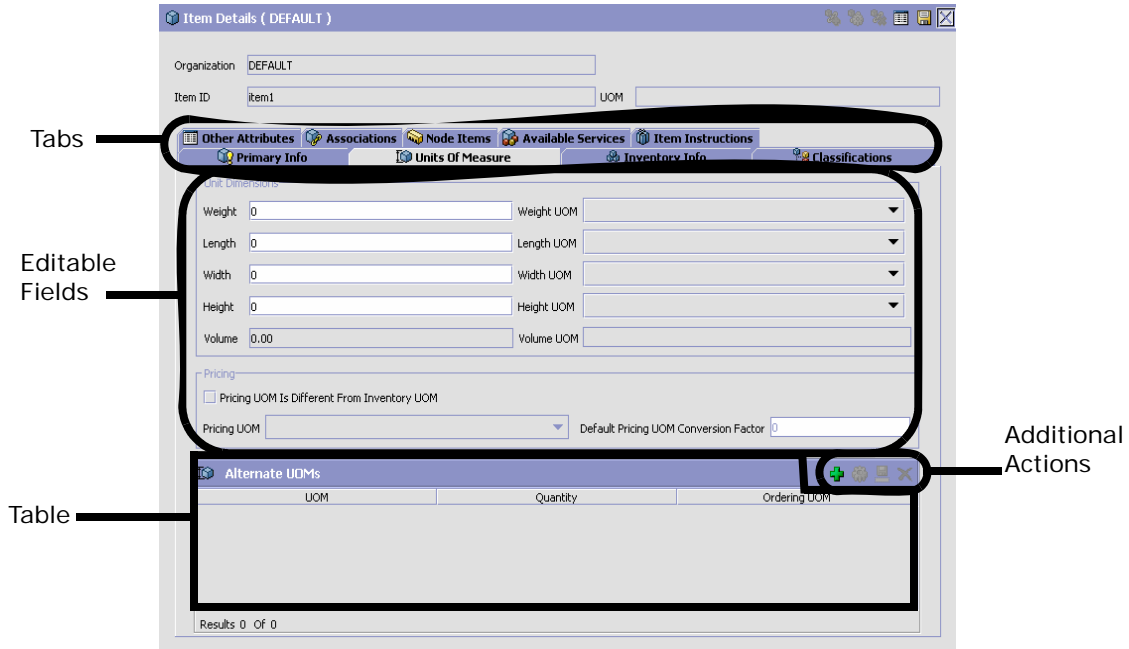
Dimension UOMs ( DEFAULT )	
UOM Code	UOM Description
CM	Centimeter
FEET	Feet
IN	Inch
KM	Kilometer
METER	Meter
MILE	Mile

Results 6 Of 6

2.2.2.3 Details Window

A details window is the main interface through which a bulk of the configuration is done. A details window can contain editable fields and tables, tabs to configure different aspects of an entity, and additional actions that can be performed on an entity.

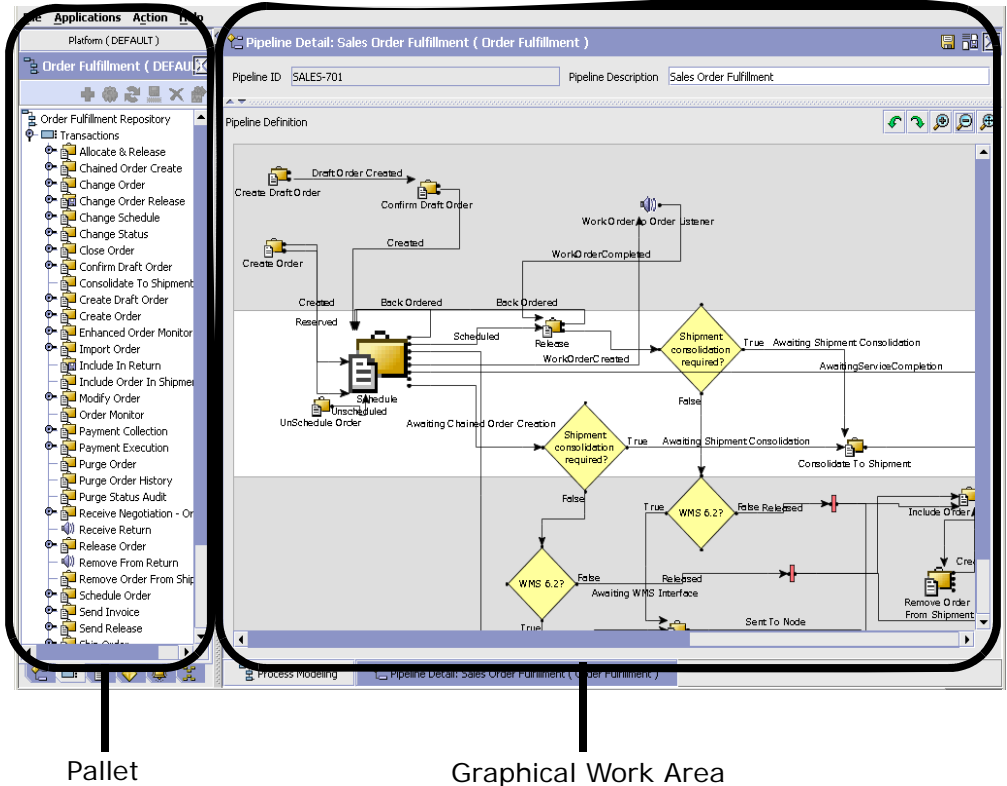
**Figure 2–10 Details Window Example**



### 2.2.2.4 Drag and Drop Window

You can use a graphical drag and drop window to ease the construction of pipelines, pipeline determination, event handlers, status monitoring rules, and services. A drag and drop window consists of a pallet and a graphical work area.

Figure 2–11 Drag and Drop Window Example



To begin building any of these entities, choose a component, such as a transaction, from the pallet. Drag the component into the graphical work area. The transaction is now displayed as a graphical representation of itself.

Many components have one or more branches. To connect the next component to the originating component, you must drag the graphical component until it forms a connecting line with one of the other component's sides, links can be set up horizontally or vertically. To delete any components or links, right-click the component and choose Delete. Once components and links have been established you can move them around by dragging them, the links redraw themselves according to the new position. If you hold CTRL while dragging a component, the component is copied within the graphical work area.

## 2.3 Actions Available Throughout the Yantra 7x Configurator

The following actions can be performed throughout the Yantra 7x Configurator:

- [Using Configurator's Lookup Functionality](#)
- [Viewing the User Logged into the Configurator](#)
- [Using Lists and List Filtering](#)
- [Using On-Line Help](#)
- [Troubleshooting Errors](#)
- [Using Special Characters](#)

### 2.3.1 Using Configurator's Lookup Functionality

Throughout the Yantra 7x Configurator there are many fields that have a lookup functionality to find or create additional records as they pertain to that field. For example, on the Primary Info tab of the Organization Details screen, the Locale field has a lookup functionality to create a new locale from that screen. When you choose the Create New lookup button the Locale Details information appears in a pop-up screen for you to modify.

**Figure 2–12 Lookup Icon Example**



The information that is displayed in a lookup field varies depending on how many records you have pertaining to that particular field. When there are 20 or less records, the lookup displays as a drop-down list with a Create New button. When there are between 21 and 75 records, the lookup displays as a drop-down list with a Search button.


When there are more than 75 records, the lookup displays as a text box with a Search button. You can type the value in the text box or search for the value using the Search button. If you enter a value, it is validated when it is saved. You should always type the value as it would appear if it was displayed as a drop-down list. For example, for a currency lookup, you should type the currency description in the text box even though the

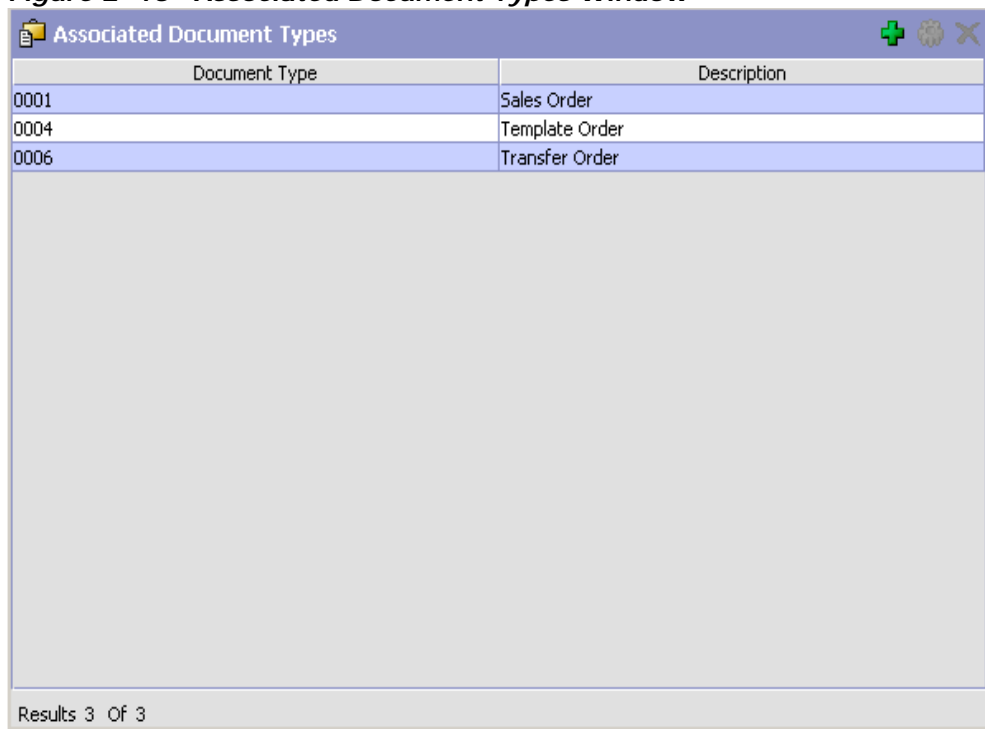
currency code is saved in the table. An error is displayed on save if the user has entered an invalid value.

When you use a lookup for a particular field in the Configurator, you should refer to the corresponding section in this guide to set up the particular information.

### 2.3.2 Viewing the Document Types Associated with an Application

In the Distributed Order Management, Supply Collaboration, Reverse Logistics, and Logistic Management configuration applications, you can view all of the document types associated with the application. Sales Order, Transfer Order, and Purchase Order are all examples of document types.

To view an application's associated document types, open the applicable application from the menu and choose  from the application rules side panel. The Associated Document Types window appears displaying a list of all of the document types associated with the application you are working in.

**Figure 2–13 Associated Document Types Window**


Document Type	Description
0001	Sales Order
0004	Template Order
0006	Transfer Order


Results 3 Of 3

### 2.3.2.1 Adding a Document Type to an Application

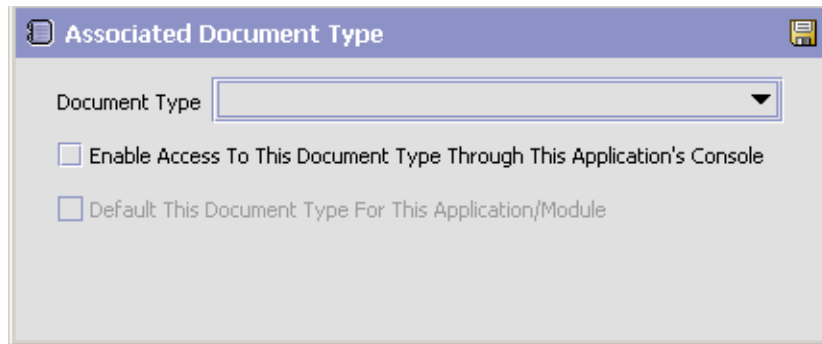
You can add a document type that is associated with another application to the application you are currently working in.


**Important:** An added document type's associated screens may be irrelevant to the application you are associating it with.

To add a document type to an application:

1. From the Associated Document Types window, choose . The Associated Document Type pop-up window appears.





2. From Document Type, select the document type that you want to associate with the application.
3. Select Enable Access To This Document Through This Applications Console.
4. Choose .

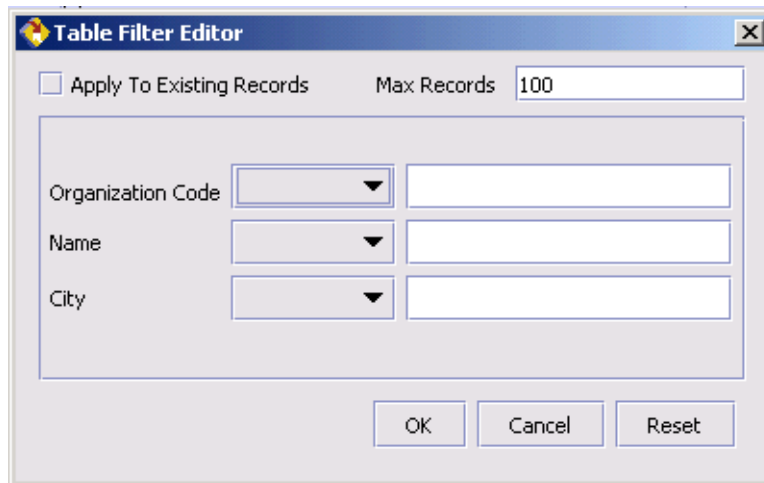
### 2.3.3 Viewing the User Logged into the Configurator

You can view the user logged into the Configurator and their locale at any time. To view this information, move your mouse over the User icon and Locale icons in the bottom right-hand corner of the application to display the tool tips.

### 2.3.4 Using Lists and List Filtering

When you perform a search in the Configurator, a list of entities is returned in a search results list based on the criteria you searched on. You can filter and arrange any information that appears in a list by right-clicking anywhere on the list's column headings and using the Table Filter Editor associated with the list.

**Figure 2–14 Table List Editor Window Example**



**Important:** When you perform a search, only 100 records are listed by default. Use the list's Table Filter Editor to increase the maximum amount of records returned by a search.

## 2.3.5 Date and Time Entry

Date fields through the Configurator have a calendar icon that can be used to find dates as it pertains to that field. When you click on this icon, a small calendar displays. You can navigate through this calendar to determine the appropriate date. For example, on the Create Calendar window, the Default Effective To field has a calendar icon that you can use to verify the appropriate ship by date to populate the field.

**Figure 2–15 Calendar Icon example**



You can also enter time of day information throughout the Configurator. To do this, double click on the time field, and enter the time of day.

**Figure 2–16 Time Field example**

Shift Name	Start Time	End Time
	<input type="text"/>	

Time should be entered in a 24 hour time format everywhere throughout the Configurator.



### 2.3.6 Using On-Line Help

You can access the Yantra 7x On-Line Help through Help > Online Help.

### 2.3.7 Troubleshooting Errors

You can view the description and cause of any error raised in Yantra 7x, as well as actions to take to troubleshoot it.

To view Yantra 7x system error descriptions:

1. From the menu bar, choose Help > Troubleshooting. The Error Search window appears.
2. Enter the applicable search criteria and choose . A list of error codes and their descriptions are displayed.
3. Choose  to view the cause of the error and action to take to troubleshoot it.

### 2.3.8 Using Special Characters

Throughout the Yantra 7x Configurator there may be instances where you need to use special characters in data entry. For information regarding the use of special characters in Yantra 7x, see the *Yantra 7x Customization Guide*.



## Configuring Cross Application Carrier Services

---

You can set up codes to identify the different carrier services a Carrier uses to ship orders.

The following are examples of different carrier service codes:



- Standard Mail
- 2nd Day Air
- Ground

You can use the Carrier Services branch for:

- [Creating a Carrier Service Code](#)
- [Modifying a Carrier Service Code](#)
- [Deleting a Carrier Service Code](#)

### 3.1 Creating a Carrier Service Code

To create a carrier service code:

1. From the tree in the application rules side panel, choose Cross Application > Carrier Services. The Carrier Services window displays in the work area.
2. Choose . The Carrier Service Details pop-up window appears.
3. Enter information in the applicable fields. Refer to [Table 3–1](#) for field value descriptions.
4. Choose .

**Carrier Service Details**

Carrier Service Code

Carrier Service Description

Advanced Transit Days

Distance Per Day

Carrier Type

Delivers On

☐ Mon ☐ Tue ☐ Wed ☐ Thu ☐ Fri ☐ Sat ☐ Sun

**Table 3–1 Carrier Service Details Pop-Up Window**


Field	Description
Carrier Service Code	Enter the name of the carrier service.
Carrier Service Description	Enter a brief description of the carrier service.
Advanced Transit Days	<p>Enter the maximum number of days that the service allows for delivery. For example, 1 Day Air would have a maximum of 1 transit day, whereas Ground may have a maximum of 5 transit days.</p> <p><b>Note:</b> This number is used for order line scheduling. This value is only used if the Use Advanced Transit Time Calculations flag on the Other Rules tab under Distributed Order Management &gt; Cross Application &gt; Logistics &gt; Logistics Attributes is selected. For more information about this field, see the <i>Yantra 7x Distributed Order Management Configuration Guide</i>.</p>
Distance Per Day	<p>Enter the maximum distance that the service travels each transit day.</p> <p><b>Note:</b> This number is used for order line scheduling. This value is only used if the Use Advanced Transit Time Calculations flag on the Other Rules tab under Distributed Order Management &gt; Cross Application &gt; Logistics &gt; Logistics Attributes is selected. For more information about this field, see the <i>Yantra 7x Distributed Order Management Configuration Guide</i>.</p>

**Table 3–1 Carrier Service Details Pop-Up Window**

Field	Description
Carrier Type	Select a carrier type from the drop-down list.
Delivers On	Check the days of the week on which the carrier service delivers.


## 3.2 Modifying a Carrier Service Code

To modify a carrier service code:

1. From the tree in the application rules side panel, choose Cross Application > Carrier Services. The Carrier Services window appears in the work area.
2. Enter information in the applicable fields. Refer to [Table 3–1](#) for field value descriptions.
3. Choose .

## 3.3 Deleting a Carrier Service Code

To delete a carrier service code:

1. From the tree in the application rules side panel, choose Cross Application > Carrier Services. The Carrier Services window appears in the work area.
2. Select the applicable carrier service code and choose .





## Configuring Cross Application Carrier Special Services

---

You can set up codes to identify the different carrier special services that a Carrier uses while shipping and delivering orders.

The following are examples for carrier special service codes:

- Adult Signature Required
- Cash on Delivery
- COD
- Contract Print Return Label
- Declared Value Insurance
- Delivery Confirmation
- Hazardous Material
- Hold At Airborne
- Hold For PickUp
- Hundred Weight
- Online Call Tag
- Saturday Delivery
- Saturday PickUp
- Ship Alert
- Ship Notification
- Signature Required


- Sunday Delivery
- Verbal Confirmation

You can use the Carrier Special Services branch for:

- [Creating a Carrier Special Service Code](#)
- [Modifying a Carrier Special Service Code](#)
- [Deleting a Carrier Special Service Code](#)

### 4.1 Creating a Carrier Special Service Code

To create a carrier special service code:

1. From the tree in the application rules side panel, choose Cross Application > Carrier Special Services. The Carrier Special Services window appears in the work area.
2. Enter information in the applicable fields. Refer to [Table 4–1](#) for field value descriptions.
3. Choose .

Carrier Special Services ( DEFAULT )	
Carrier Special Services Code	Description
1ZASD	Air Service Document
1ZCOD	Collect on Delivery
ADULTSIGN	Adult Signature Required
ASD	Air Service Document
CONPRLBL	Contract Print Return Label
DECLVAL	Declared Value Insurance
DELCONF	Delivery Confirmation
HLDPRPKUP	Hold For Pickup
ONLCALLTAG	Online Call Tag
SATDELI	Saturday Delivery
SHIPNTFY	Ship Notification
SIGNREQD	Signature Required
STDCALLTAG	Standard Call Tag
TAGLESSCOD	Tagless COD
VRBLCONF	Verbal Confirmation
*	
Results 15 Of 15	


**Table 4–1 Carrier Special Services Window**

Field	Description
Carrier Special Services Code	Enter the name of the carrier special service.
Description	Enter a brief description of the carrier special service.

## 4.2 Modifying a Carrier Special Service Code


To modify a carrier special service code:

1. From the tree in the application rules side panel, choose Cross Application > Carrier Special Services. The Carrier Special Services window appears in the work area.

2. Modify information in the applicable fields. Refer to [Table 4–1](#) for field value descriptions.
3. Choose .

### 4.3 Deleting a Carrier Special Service Code

To delete a carrier special service code:

1. From the tree in the application rules side panel, choose Cross Application > Carrier Special Services. The Carrier Special Services window appears in the work area.
2. Select the applicable carrier special service code and choose .


## Configuring Transportation

---

You can record transportation or in-transit updates for a container. These updates are recorded against activities defined in Yantra 7x as transportation activities.

### 5.1 Creating an Activity Code

To create an activity code:

1. From the tree in the application rules side panel, choose Transportation > Activities. The Activity Search window displays.
2. Enter information in the applicable fields. Refer to [Table 5–1](#) for field value descriptions.
3. Choose .

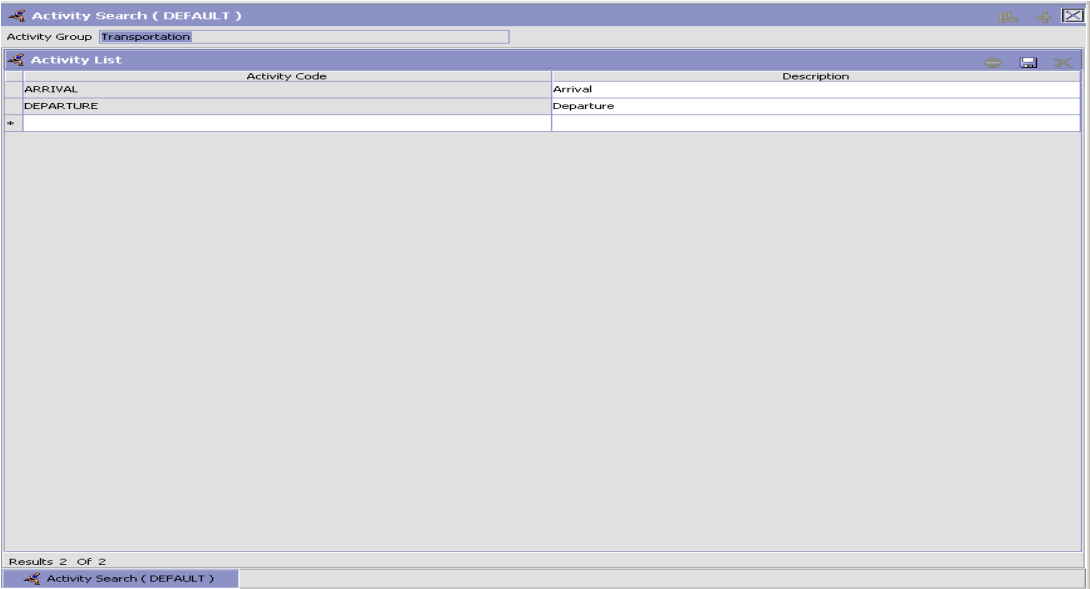


Table 5–1 Activity Search Window

Field	Description
Activity Group	This field is automatically populated by the system as 'Transportation'.
Activity List	
Activity Code	Enter a name for the activity code. This Activity Code is the unique identity of the activity.
Description	Enter a brief description for the activity code.

5.2 Modifying an Activity Code

To modify an activity code:

1. From the tree in the application rules side panel, choose Transportation > Activities. The Activity Search window displays, with the list of activities.
2. Enter information in the applicable fields. Refer to Table 5–1 for field value descriptions.

3. Choose .

---

---


**Note:** It is recommended that you do not modify the list of activity codes provided by Yantra 7x WMS.

---

---

## 5.3 Deleting an Activity Code

To delete an activity code:

1. From the tree in the application rules side panel, choose Transportation > Activities. The Activity Search window displays, with the list of activities.
2. Choose the Activity Code to be deleted.
3. Choose .

---

---

**Note:** It is recommended that you do not modify the list of activity codes provided by Yantra 7x WMS.

---

---





# 6

## Configuring a Document's Modification Reasons

---

You can define common codes for **modification reasons**. These codes define why a modification was made by a user in the Application Consoles.


**Note:** In addition to modification reasons, the codes that you define are used as hold reasons when you put an order on hold in the Application Consoles.

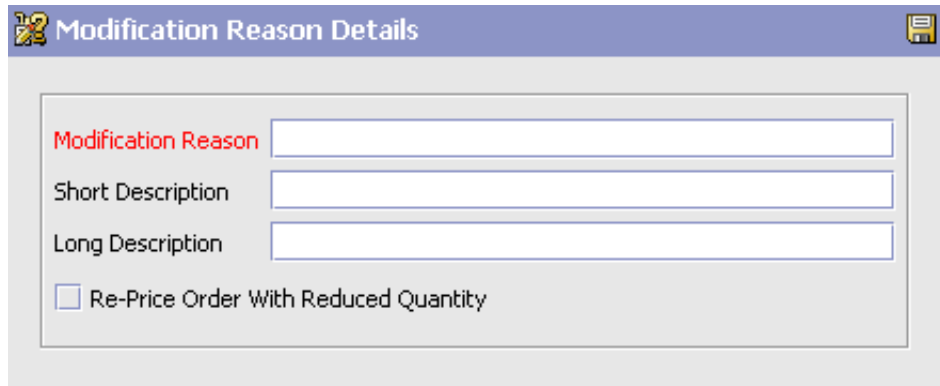
You can use the Modification Reasons branch for:

- [Creating a Modification Reason](#)
- [Modifying a Modification Reason](#)
- [Deleting a Modification Reason](#)

### 6.1 Creating a Modification Reason

To create a modification reason:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Modification Reasons. The Modification Reasons window appears in the work area.
2. Choose . The Modification Reason Details pop-up window appears.



**Modification Reason Details**

Modification Reason

Short Description

Long Description

☐ Re-Price Order With Reduced Quantity

3. In Modification Reason, enter the modification reason.
4. In Short Description, enter a brief description of the modification reason.
5. In Long Description, enter a more detailed description of the modification reason.
6. If this modification reason requires that the order be re-priced due to a reduced quantity, check the Re-Price Order With Reduced Quantity checkbox.

This flag is applicable only if this modification reason is used for cancellations, where re-pricing needs to occur against a reduced quantity: the quantity against which the order line is re-priced (re-pricing quantity) is adjusted to the reduced quantity. For more information on re-pricing quantity, refer to the *Yantra 7x Javadocs*.


**Note:** If this modification reason is used for a modification which does not reduce quantity, this flag is not applicable.

**Note:** This field does not exist for Load Modification Reasons.

7. Choose .

## 6.2 Modifying a Modification Reason

To modify a modification reason:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Modification Reasons. The Modification Reasons window appears in the work area.
2. Select the applicable modification reason and choose . The Modification Reason Details pop-up window appears.
3. In Short Description, enter a brief description of the modification reason.
4. In Long Description, enter a more detailed description of the modification reason.
5. If this modification reason requires that the order be re-priced due to a reduced quantity, check the Re-Price Order With Reduced Quantity checkbox.

This flag is applicable only if this modification reason is used for cancellations, where re-pricing needs to occur against a reduced quantity: the quantity against which the order line is repriced (re-pricing quantity) is adjusted to the reduced quantity. For more information on re-pricing quantity, refer to the *Yantra 7x Javadocs*.

**Note:** If this modification reason is used for a modification which does not reduce quantity, this flag is not applicable.


**Note:** This field does not exist for Load Modification Reasons.

6. Choose .

## 6.3 Deleting a Modification Reason

To delete a modification reason:

## Deleting a Modification Reason

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Modification Reasons. The Modification Reasons window appears in the work area.
2. Select the applicable modification reason and choose .

# Configuring a Load Document's Pipeline

---

**Important:** Be aware that return fulfillment requires sourcing configuration. Sourcing configuration is accessible through the Distributed Order Management configuration grouping. For more information about configuring sourcing, see the *Yantra 7x Distributed Order Management Configuration Guide*.

To complete a load document's lifecycle, each document has a set of different processes that it can go through. These processes are called process types.

- Receipt

You can configure the rules and components specific to a load document's process type.

You can use process type configuration for:

- [Defining Process Type Details](#)
- [Process Type Pipeline Configuration](#)

## 7.1 Defining Process Type Details

You can define the parameters and templates that distinguish a process type.

For more information about defining process type details, see *Yantra 7x Platform Configuration Guide*.

## 7.2 Process Type Pipeline Configuration

A **process type pipeline** is a series of transactions and statuses that guide document types, such as a Sales Order, through a predefined process. A pipeline consists of the different statuses a document goes through during fulfillment, negotiation, shipment, or receipt. You can also set up transactions consisting of events, actions, and conditions, as they pertain to the pipeline you are configuring.

### Repositories

A repository is a logical collection of entities that define the business process workflow.

The following entities are included in a repository:

- Pipelines
- Transactions
- Statuses
- Conditions
- Actions
- Services

Yantra 7x provides a base repository for each of the system defined process types. Some of the entities within a repository are copied when creating a new document type. For more information about creating a new document type, see the *Yantra 7x Platform Configuration Guide*.

The load process is modeled through a pipeline. This represents the process configuration that is unique to an organization. An organization may also specify unique processes for each participating Enterprise.

### 7.2.1 Defining Pipeline Determination

**Pipeline determination** is used to set up conditions that affect which pipeline is used during the start of the business process workflow. For example, an organization deals with sales orders that sometimes contain hazardous materials. They have two separate pipelines, one in which orders with order lines without any hazardous materials go through and one in which orders with order lines containing hazardous materials must go through for inspection before continuing through the order process.

The organization uses pipeline determination to set up a condition that determines whether or not order lines contain hazardous materials and sends the order line down the correct pipeline.

When you expand the Pipeline Determination branch, the components displayed depends on what role you are logged in as. If you are logged in as a Hub role, the Hub Rule is displayed. If you are logged in as an Enterprise role, both the Hub Rule and My Rule components are displayed. Double-click on the applicable node to display the pipeline determination rules.

---

---

**Note:** If you are logged in as an Enterprise role, the Hub Rule screen is grayed out and cannot be modified.

---

---

Drag conditions and pipelines into the work area to construct pipeline determination rules. A single pipeline or condition must be the root. Conditions cannot link back to an earlier component in the chain and a pipeline cannot be linked to twice.

---

---

**Note:** When configuring pipeline determination for an order document type pipeline, please note that pipeline determination is only considered when adding a line or creating an order. When changes are made to draft orders pipeline determination does not occur.

---

---

### 7.2.1.1 Condition Variables for Pipeline Determination

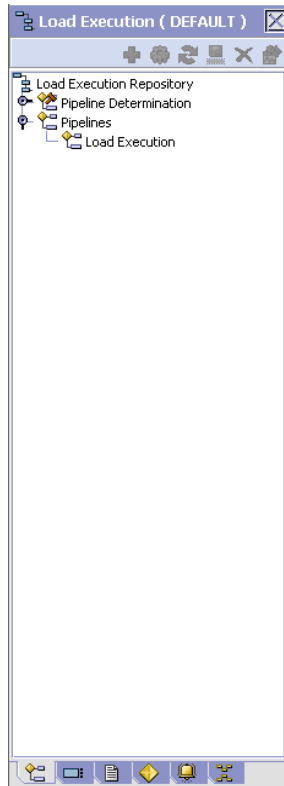
For a list of the condition variables that can be used for pipeline determination, refer to [Appendix B, "Condition Builder Attributes"](#).

## 7.2.2 Pipelines

For more information about configuring pipelines, see the *Yantra 7x Platform Configuration Guide*.

To view the load execution pipeline details:

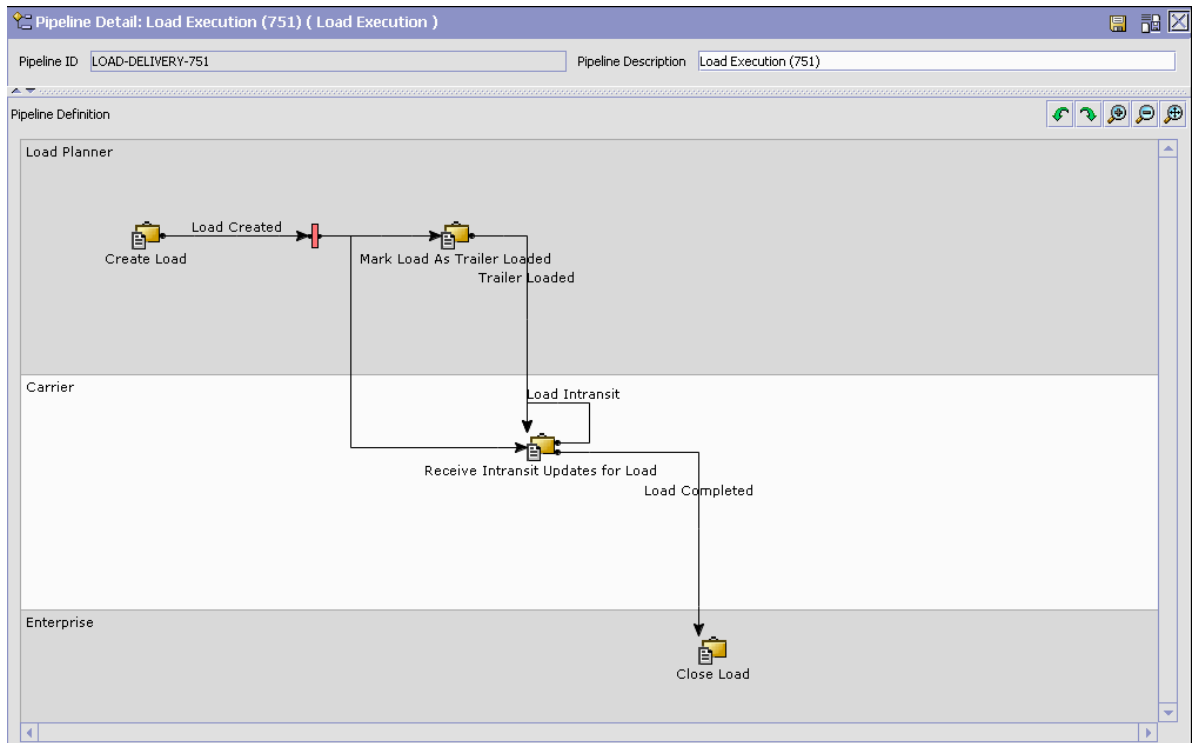
1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Load Execution Process Model. The Load Execution window is displayed.



2. In the Load Execution window, choose Load Execution Repository > Pipelines > Load Execution.
3. The Pipeline Detail: Load Execution (Load Execution) window is displayed.

For more information about creating and modifying a pipeline, see the *Yantra 7x Platform Configuration Guide*.






### 7.2.3 Transactions

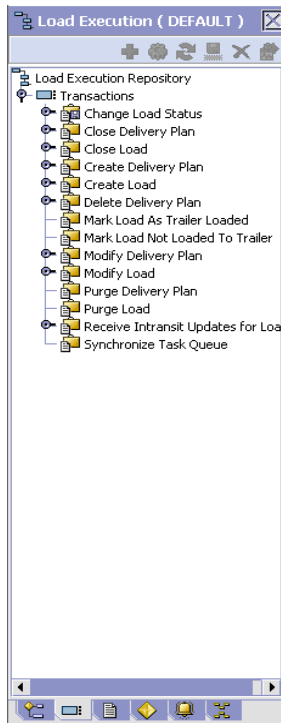
Every process type has a set of base transactions defined for it. A transaction is a logical unit of work that is necessary for performing activity within Yantra 7x. Base transactions are predefined transactions that contain information about how the transaction behaves, such as how many copies of a transaction can be kept in a process type and whether or not it can have configurable base pick and drop statuses. Base transactions can be used to create new transactions. These transactions can be changed within the limits defined in the base transaction.

For more information about transactions, see the *Yantra 7x Platform Configuration Guide*.

To view the transaction details for a load execution pipeline:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Load Execution Process Model. The Load Execution window is displayed.
2. In the Load Execution window, choose .
3. The Transactions tab window is displayed.

For more information about creating and modifying transactions, see the *Yantra 7x Platform Configuration Guide*.



**Table 7–1 Load Execution Pipeline - Transactions Tab Window**

Field	Description
Change Load Status	This transaction represents any modifications that may be made involving a load's status.
Close Delivery Plan	This transaction represents a delivery plan being closed and ready to be purged from the system.
Close Load	This transaction represents a load being closed and ready to be purged from the system.
Create Delivery Plan	This transaction represents the creation of a delivery plan.
Create Load	This transaction represents a the creation of a load.
Delete Delivery Plan	This transaction represents the deletion of a delivery plan.
Mark Load As Trailer Loaded	This transaction indicates that all the shipments belonging to a load have been loaded onto the appropriate Carrier.
Mark Load Not Loaded To Trailer	This transaction indicates that a load has not been loaded onto the appropriate Carrier.
Modify Delivery Plan	This transaction represents modifications made to a delivery plan.
Modify Load	This transaction represents modifications made to a load.
Purge Delivery Plan	This transaction represents the process of removing delivery plans from the system.
Purge Load	This transaction represents the process of removing loads from the system.
Receive Intransit Updates for Load	This transaction represents the process of receiving any updated concerning the status of a load while it is in transit.
Synchronize Task Queue	This transaction represents the process of synching the load execution task queue.


## 7.2.4 Statuses

**Statuses** are the actual states that a document moves through in the pipeline. A transaction can contain two types of statuses, a drop status

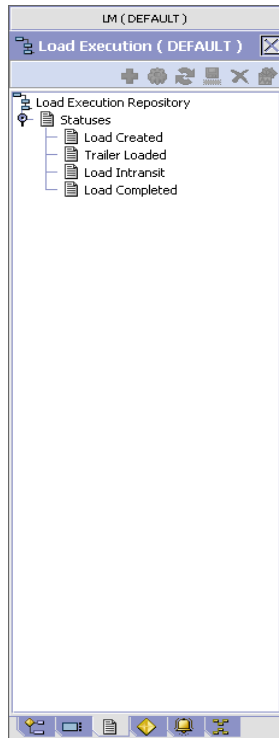
and a pickup status. A document is moved into a **drop status** when the events and conditions of a transaction have been completed. A **pickup status** takes the document from the previous drop status and moves it through the next transaction. Created and Scheduled are examples of statuses.

For more information about statuses, see the *Yantra 7x Platform Configuration Guide*.

To view the status details of a load execution pipeline:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Load Execution Process Model. The Load Execution window is displayed.
2. In the Load Execution window, choose .
3. The Statuses tab window is displayed.

For more information about creating and modifying statuses, see the *Yantra 7x Platform Configuration Guide*.



**Table 7–2 Load Execution Pipeline - Statuses Tab Window**


Field	Description
Load Created	This indicates that a load has been created.
Trailer Loaded	This indicates that a load has been loaded on to the appropriate Carrier.
Load Intransit	This indicates that a load is physically in between stops configured in the delivery plan.
Load Completed	This indicates that a load has been delivered to the applicable stop or destination as per the delivery plan.

## 7.2.5 Conditions

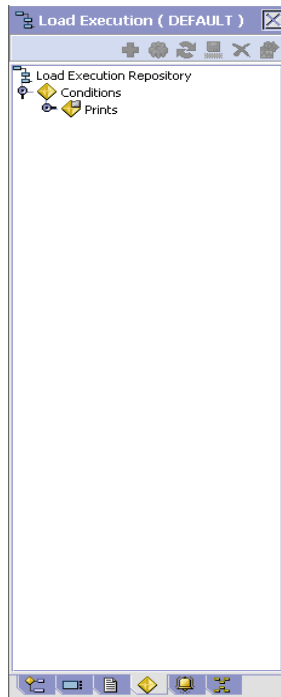
A **condition** matches document type attributes against decision points and routes the documents to different paths based on the specified attribute and value combinations. The document type attributes against which conditions can be created are predefined in Yantra 7x. You can use these attributes in any combination or you can create conditions that execute the appropriate application logic for specific circumstances.

For more information about conditions, see the *Yantra 7x Platform Configuration Guide*.

To view the condition details of a load execution pipeline:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Load Execution Process Model. The Load Execution window is displayed.
2. In the Load Execution window, choose .
3. The Conditions tab window is displayed.

For more information about creating and modifying conditions, see the *Yantra 7x Platform Configuration Guide*.




## 7.2.6 Actions

An **action** is a process or program that is triggered by an event. These processes and programs send user alert notifications and automatically resolve issues.

For example, when an order is released (the event), you can set an action to send the customer an e-mail.

For more information about actions, see the *Yantra 7x Platform Configuration Guide*.

To view the action details of an outbound shipment pipeline:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Load Execution Process Model. The Load Execution window is displayed.
2. In the Load Execution window, choose .

3. The Actions tab window is displayed.

For more information about creating and modifying actions, see the *Yantra 7x Platform Configuration Guide*.



## Configuring a Document's Instruction Types

---

You can define the common codes used when adding special instructions to an order document.

The following are Yantra 7x's default instruction types:


- PICK
- PACK
- SHIP
- GIFT
- OTHER

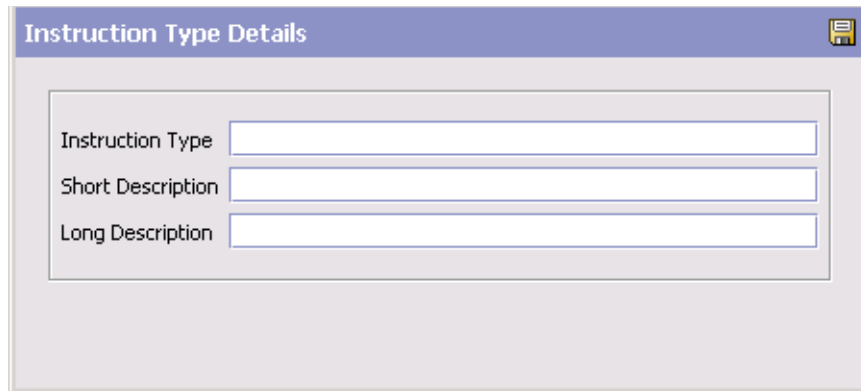
You can use the Instruction Types branch for:

- [Creating an Instruction Type](#)
- [Modifying an Instruction Type](#)
- [Deleting an Instruction Type](#)


### 8.1 Creating an Instruction Type

To create an instruction type:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Instruction Types. The Instruction Types window appears in the work area.
2. Choose . The Instruction Type Details pop-up window appears.





The screenshot shows a dialog box titled "Instruction Type Details". It contains three text input fields: "Instruction Type", "Short Description", and "Long Description". Each field is preceded by its label. The dialog box has a standard Windows-style title bar with a save icon in the top right corner.

3. In Instruction Type, enter the instruction type.
4. In Short Description, enter a brief description of the instruction type.
5. In Long Description, enter a more detailed description of the instruction type.
6. Choose .


## 8.2 Modifying an Instruction Type

To modify an instruction type:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Instruction Types. The Instruction Types window appears in the work area.
2. Select the applicable instruction type and choose . The Instruction Type Details pop-up window appears.
3. In Short Description, enter a brief description of the instruction type.
4. In Long Description, enter a more detailed description of the instruction type.
5. Choose .

## 8.3 Deleting an Instruction Type

To delete an instruction type:

1. From the tree in the application rules side panel, choose Document Specific > (*Document Type*) > Instruction Types. The Instruction Types window appears in the work area.
2. Select the applicable instruction type and choose .



# Configuring a Load Document's Charge Categories

---

You can define **charge definitions** that you can associate with orders and invoices through charge categories. These categories contain a group of related charge names that can be used when the particular category is used.

The following are Yantra 7x default charge definitions:

- Shipping
- Handling
- Personalization
- Discount

**Note:** Do not use charge definitions to create tax structures.


Use the Charge Categories branch for:

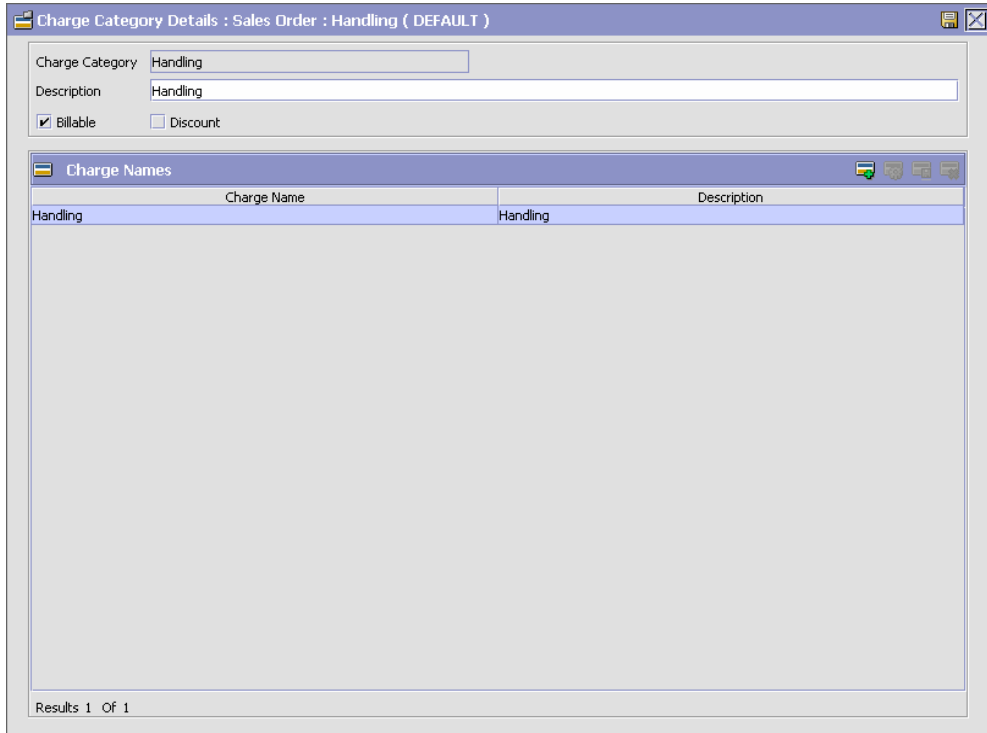
- [Creating a Charge Category](#)
- [Modifying a Charge Category](#)
- [Deleting a Charge Category](#)

## 9.1 Creating a Charge Category


To create a charge category:

## Creating a Charge Category

1. From the tree in the application rules side panel, choose Document Specific > Load > Charge Categories. The Charge Categories window appears in the work area.
2. Choose . The Charge Category Details window is displayed.



Charge Name	Description
Handling	Handling

3. In Charge Category, enter the name of the charge category.
4. In Description, enter a brief description of the charge category.
5. Choose .

**Note:** Charge categories cannot be localized. For more information about localization, see the *Yantra 7x Localization Guide*.


You can use the Charge Category Details window for:

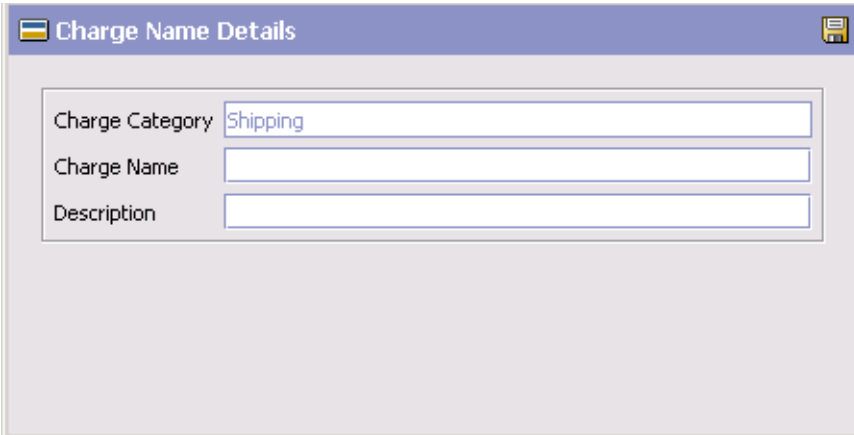
- Adding a Charge Name Associated with a Charge Category
- Modifying a Charge Name Associated with a Charge Category
- Deleting a Charge Name Associated with a Charge Category

### 9.1.1 Adding a Charge Name Associated with a Charge Category


Charge names are names of the actual charges included in the charge definition.

To add a charge name to a charge category:

1. In the Charge Category Details window, choose . The Charge Name Details pop-up window is displayed.





The screenshot shows a window titled "Charge Name Details". Inside the window, there are three text input fields. The first field is labeled "Charge Category" and contains the text "Shipping". The second field is labeled "Charge Name" and is empty. The third field is labeled "Description" and is empty.

2. In Charge Name, enter the charge name.
3. In Description, enter a brief description of the charge name.
4. Select Billable if the charge is billable. Non-billable charges are not considered in order totals, but do appear in invoices.
5. Select Discount if the charge you are creating is a discount charge type.
6. Choose .


**Note:** Charge names cannot be localized. For more information about localization, see the *Yantra 7x Localization Guide*.

### 9.1.2 Modifying a Charge Name Associated with a Charge Category

To modify a charge category's charge name:



1. In the Charge Category Details window, select the applicable charge name and choose . The Charge Name Details pop-up window is displayed.
2. In Description, enter a brief description of the charge name.
3. Select Is Billable if the charge is billable. Non-billable charges are not considered in order totals, but do appear in invoices.
4. Select Is Discount if the charge you are creating is a discount charge type.
5. Choose .

### 9.1.3 Deleting a Charge Name Associated with a Charge Category

To delete a charge category's charge name select the applicable charge name in the Charge Category Details window and choose .

## 9.2 Modifying a Charge Category


To modify a charge category:

1. From the tree in the application rules side panel, choose Document Specific > Load > Charge Categories. The Charge Categories window appears in the work area.
2. Select the applicable charge category and choose . The Charge Category Details window is displayed.
3. In Description, enter a brief description of the charge category.
4. Choose .



## 9.3 Deleting a Charge Category

To delete a charge definition:

1. From the tree in the application rules side panel, choose Document Specific > Load > Charge Categories. The Charge Categories window appears in the work area.
2. Select the applicable charge category and choose .



# 10

## Configuring a Load Document's Purge Criteria

**You can** set qualifications around each type of purge. A **purge** is the process by which old data is removed from the system database. Purges minimize the number of unused database records to increase search efficiency and reduce the size of the required physical disk. In Purge Criteria, default purge rules are provided. These can be modified for your system operations.


[Table 10–1](#) lists the purge rules provided for load document types in Yantra 7x.


*Table 10–1 Load Document Type Purge Rules*



Rule	Description	Retention Days
DELIVERYPLANPRG	Purges delivery plan information.	30
LOADPRG	Purges load information.	30

### 10.1 Modifying a Load Document Type's Purge Criteria Rule

To modify a load document type's purge criteria rule:

1. From the tree in the application rules side panel, choose Document Specific > Load > Purge Criteria. The Purge Criteria window appears in the work area.
2. Select the applicable purge criteria rule and choose . The Purge Criteria Details pop-up window is displayed.



- 3. Enter information in the applicable fields. Refer to [Table 10–2](#) for field value descriptions.
- 4. Choose .

Purge Criteria Details

Purge CodeORDERHISTPRGDescriptionOrder History Purge

Rollback SegmentRetention Days30

☒ Write To Log FileLog File NameOrder\_history\_purge\_data.log

Additional Purge Criteria

Line Type	Additional Retention Days
[E1 line type]	600
*	

Table 10–2 Purge Criteria Details Pop-up Window

Field	Description
Purge Code	Identifies a purge program. This is a system defined code.
Description	Describes the type of purge.
Rollback Segment	Defines the rollback segment that should be explicitly used for the purge transaction qualified by the purge code. This is useful when there are huge logical data sets that have to be purged. This is optional and used for order related purges.
Retention Days	Enter the number of days of data to be retained in the database (going backwards from the time the program runs). Make sure that your table size takes into account the number of retention days entered here. The inventory purge does not take retention days into account when purging.

**Table 10–2** *Purge Criteria Details Pop-up Window*

Field	Description
Write to Log File	Select this field if you want purged data written to a log. The log can be backed up and used as a journal at a later date.
Log File Name	<p>Enter a log file name. The log file is created in the directory specified in the <code>yfs.purge.path</code> of <code>yfs.properties</code>. If a variable is introduced, then <code>yfs.purge.path</code> is ignored. For more information on using variables for the log file directory refer to <i>Yantra 7x Platform Configuration Guide</i>.</p> <p>For information about filename limitations related to internationalization, see the <i>Yantra 7x Localization Guide</i>.</p>



## Configuring a Load Document's Load Types

---

You can define codes for load types that appear on a load document. A load carries one or more shipments from one point to another point.


**Note:** This code has no application logic associated with it and can be set up as per your business practices.

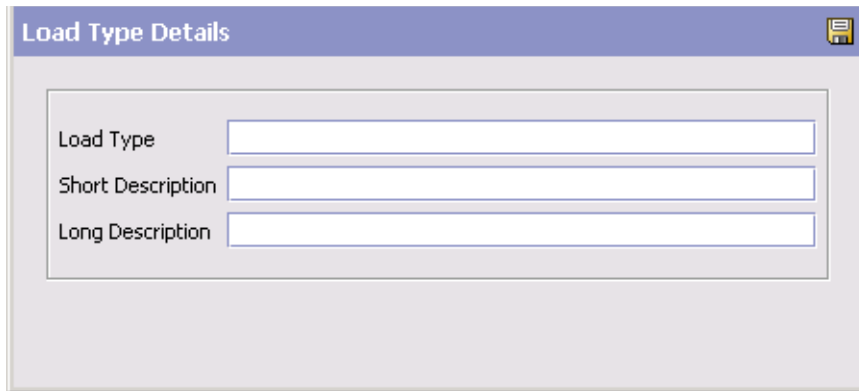
You can use the Load Type branch for:

- [Creating a Load Type](#)
- [Modifying a Load Type](#)
- [Deleting a Load Type](#)


### 11.1 Creating a Load Type

To create a load type:

1. From the tree in the application rules side panel, choose Document Specific > Load > Load Type. The Load Type window appears in the work area.
2. Choose . The Load Type Details pop-up window appears.





The screenshot shows a dialog box titled "Load Type Details". It contains three text input fields with the following labels: "Load Type", "Short Description", and "Long Description". The dialog box has a standard Windows-style title bar with a small icon in the top right corner.

3. In Load Type, enter the name of the load type.
4. In Short Description, enter a brief description of the load type.
5. In Long Description, enter a more detailed description of the load type.
6. Choose .

## 11.2 Modifying a Load Type


To modify a load type:

1. From the tree in the application rules side panel, choose Document Specific > Load > Load Type. The Load Type window appears in the work area.
2. Select the applicable load type and choose . The Load Type Details pop-up window appears.
3. In Short Description, enter a brief description of the load type.
4. In Long Description, enter a more detailed description of the load type.
5. Choose .

## 11.3 Deleting a Load Type

To delete a load type:



1. From the tree in the application rules side panel, choose Document Specific > Load > Load Type. The Load Type window appears in the work area.
2. Select the applicable load type and choose .



# 12

## Configuring a Load Document's Stop Types

---

You can modify codes for stop types that appear on a load document. A stop is a location where a shipment is picked up or dropped off.



**Note:** This code has no application logic associated with it and can be set up as per your business practices.

The following are Yantra 7x default stop types:

- D - Destination
- I - Intermediate
- O - Origin

### 12.1 Modifying a Stop Type

To modify a stop type:

1. From the tree in the application rules side panel, choose Document Specific > Load > Stop Type. The Stop Type window appears in the work area.
2. Select the applicable stop type and choose . The Stop Type Details pop-up window appears.
3. In Short Description, enter a brief description of the stop type.
4. In Long Description, enter a more detailed description of the stop type.
5. Choose .



## Configuring a Load Document's Modification Rules

---

Most load document types flow through a pipeline without requiring any intervention. However, there are times when modifications are required. Yantra 7x supports modifications through the Yantra 7x Application Consoles. It is critical for you to decide which modifications are allowed for each modification type, modification level, and status combination.

**Important:** Contemplate the business and system integration implications before turning on a setting that is disallowed as part of the system defaults.

Modification type indicates the type of modification carried out on the load document type. Yantra 7x provides the ability to perform modifications on specific attributes. An example of a modification type is adding a stop to a load's delivery plan.

Modification level indicates the level at which a particular modification type is carried out. The load document type has a modification level of Load. Modifications are applied to a particular level and a particular processing status.

The following are the default order modification types and their associated modification level:

**Table 13–1 Load Document Modification Types**

Modification Type	Description	Modification Level
Add Instruction	An instruction can be added to a load document.	Load
Add Shipment	A shipment can be added to a load document.	Load
Add Stop	A stop can be added to a load document.	Load
Add/Remove Additional Date	Additional dates that can be associated with a load document can be added and removed.	Load
Add/Remove Charge	A charge can either be added to or removed from an load document.	Load
Change Instruction	<p>A modification can be made to an instruction associated with a load document.</p> <p>The following instruction fields can be modified when this modification type is allowed:</p> <ul style="list-style-type: none"><li>• Instruction Type</li><li>• Text</li><li>• URL</li></ul>	Load
Change Other Attributes	A modification can be made to fields that do not have system or user-defined modification types associated with them.	Load
Change Other Relationships	Not used in this version.	Load
Delete Load	A load document can be deleted.	Load
Delete Stop	A stop can be deleted from a load document.	Load

**Table 13–1 Load Document Modification Types**

Modification Type	Description	Modification Level
Modify Stop	A modification can be made to a stop associated with a load document.	Load
Remove Shipment	A shipment can be removed from a load document.	Load
Resequence Stop	A load document's stops can be reordered.	Load

You can group modifications in the window by modification type, modification level, or status, by selecting the corresponding grouping from Group By. The Modification Rules window then displays the grouping you have chosen in a hierarchical structure.

**Table 13–2 Load Document Type Rule Modifications**

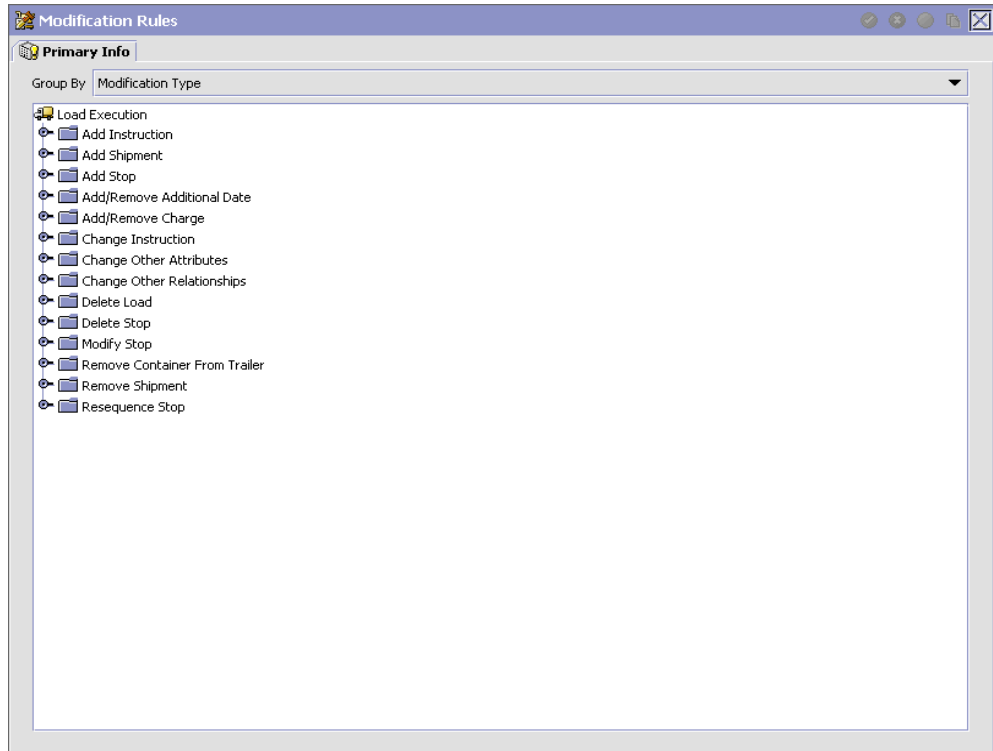
Field	Description
Status	Indicates each status that is applicable to a modification level and type.
Allow	Indicates whether or not modifications may be made at this modification level and type for the specified status.
Disallow	Indicates that no modifications may be made at this modification level and type for the specified status.
Ignore	Indicates that modifications are ignored at this modification level and type for the specified status.

**Note:** Any permissions set up for user groups override configured modification rules. For more information about configuring user group permissions, see the *Yantra 7x Platform Configuration Guide*.

## 13.1 Changing a Load Document Type's Modification Rules

To change a load document type's modification rules:

1. From the tree in the application rules side panel, choose Document Specific > Load > Modification Rules. The Modification Rules window appears in the work area.



2. Expand the applicable modification types and levels for which you want to set up rules.
3. Right click on the applicable rule and choose allow, disallow, or ignore as per your business practices. Refer to [Table 13–2](#) for field value descriptions.



## Configuring Logistics Components

---

**You can configure the components used by different logistics related functionality throughout the business application module.**

You can use the Logistics branch for:

- [Defining Logistics](#)
- [Defining Shipment Planning](#)

### 14.1 Defining Logistics

**You can** define rules and common codes associated logistics of shipping an order.

You can use the Logistics branch for:

- [Defining Freight Terms](#)
- [Defining Shipment Planning](#)

#### 14.1.1 Defining Freight Terms

You can define common codes used when associating a freight term to a Carrier. A **freight term** identifies how transportation costs are calculated.

The following are Yantra 7x default freight terms:

- Collect (COLLECT) - The buyer owns the freight payment responsibility.
- Prepaid (PREPAID) - The seller owns the freight payment responsibility.



- Third Party Collect (TP-COLLECT) - A third party organization is responsible for all or part of the payment processing, but the buyer is still responsible for the actual payment.
- Third Party Prepaid (TP-PREPAID) - A third party organization is responsible for all or part of the payment processing, but the seller is still responsible for the actual payment.

You can use the Freight Terms tab for:

- [Creating a Freight Term](#)
- [Modifying a Freight Term](#)
- [Deleting a Freight Term](#)

### 14.1.1.1 Creating a Freight Term

To create a freight term:

1. From the tree in the application rules side panel, choose Logistics Management > Freight Terms. The Freight Terms window appears in the work area.
2. Choose . The Freight Terms Details pop-up window is displayed.
3. Enter information in the applicable fields. Refer to [Table 14–1](#) for field value descriptions.
4. Enter Choose .

Freight Terms Details

Freight Terms

Short Description

Long Description

☒ Consider Buyer's Routing Guide

Charges paid by

☒ Buyer

☐ Shipper

**Table 14–1 Freight Terms Details**



Field	Description
Freight Terms	Enter the name of the freight term.
Short Description	Enter a brief description of the freight term.
Long Description	Enter a more detailed description of the freight term.
Consider Buyer's Routing Guide	<p>Both the Buyer and the Enterprise can establish routing guides, rules for shipping, and ESP (Economic Shipping parameters), which control how items are shipped. In some cases only the Buyer organization has established values for these rules, in other cases, only Enterprise has established values for these rules. If neither is set, then Hub rules are used.</p> <p>In cases where both the Buyer and the Enterprise have set values for these rules, this setting determines whether to apply Buyer's rules before applying the routing rules of the Enterprise. See <i>Yantra 7x Product Concepts</i> for more information about these shipping concepts.</p>
Charges paid by	

**Table 14–1 Freight Terms Details**

Field	Description
Buyer	Select if Buyer pays shipping charges. The system considers the freight term as "Collect".
Shipper	Select if Shipper pays shipping charges. The system considers the freight term as "Prepaid".


#### 14.1.1.2 Modifying a Freight Term

To modify a freight term:

1. From the tree in the application rules side panel, choose Logistics Management > Freight Terms. The Freight Terms window appears in the work area.
2. Select the applicable freight term and choose . The Freight Terms Details pop-up window is displayed.
3. Enter information in the applicable fields. Refer to [Table 14–1](#) for field value descriptions.
4. Choose .

#### 14.1.1.3 Deleting a Freight Term

To delete a freight term:

1. From the tree in the application rules side panel, choose Logistics Management > Freight Terms. The Freight Terms window appears in the work area.
2. Select the applicable freight term and choose .

### 14.1.2 Defining Shipment Planning


Shipment Planning is used to describe conditions that control how shipping is done. These include whether certain items can be shipped together, whether to use Economic Shipping Parameters, and how routing is performed.

You can also use Shipment Planning for:

- [Creating a Routing Guide](#)

- [Modifying a Routing Guide](#)
- [Deleting a Routing Guide](#)

To define shipping planning:

1. From the tree in the application rules side panel, choose Logistics Management > Shipment Planning. The Shipment Planning window appears in the work area.
2. Enter information in the applicable fields. Refer to [Table 14–2](#) for field value descriptions.
3. Choose .

**Figure 14–1 Shipment Planning Window**

**Shipment Planning ( DEFAULT )**

Do not mix in a shipment:

☒ Customer PO #      ☒ Department Code

☒ Mark For      ☒ Order #

☐ Order Type

Transportation optimization

Economic shipping parameters maintained ☒

Delay shipment by not more than  Days

Consolidate upto weight threshold of  LBS

Consolidate upto volume threshold of  CIN

Region Schema For Routing:

Routing Guide:

☐ Not Maintained      ☐ Maintained in Yantra      ☒ Maintained Externally

**Routing Guides**

Name	Routing Guide #	Effective Date	Freight Terms
Results 0 Of 0			

Freight Terms ( DEFAULT )      Shipment Planning ( DEFAULT )

**Table 14–2 Shipment Planning Window**

Field	Description
Do not mix in Shipment	<p>If any of the following are selected, separate shipments must be created for items that have different values for these attributes.</p> <p>For example, if Department Code is selected, items that are for different departments can not be included in the same shipment.</p>
Customer PO #	Customer's Purchase Order number.

**Table 14–2 Shipment Planning Window**

Field	Description
Mark For	Person for whom this shipment is marked for
Department Code	The department for which the item is intended.
Order #	The order number.
Order Type	The order type.
Transportation optimization	
Economic shipping parameters maintained	<p>Economic Shipping Parameters (ESP) are used in shipping consolidation. Select this field to enable the following Economic Shipping Parameters fields.</p> <p>ESP support consolidation of shipments until a weight or volume threshold is met, or until an certain time elapses. By consolidating shipments, shipping costs can be reduced</p> <p>For example, you can set that shipments should be consolidated until the shipment weight is 300 pounds, or 50 cubic feet in volume. To ensure that eventually the shipment is set, you can establish a maximum number of days to wait until the conditions are met.</p> <p>When either the weight, volume or delay shipment threshold is met, the shipment is moved to the next stage in shipping.</p>
Delay shipment by not more than ___ Days	<p>Enter the number of days this shipment can be delayed before it should be shipped.</p> <p>For example, if a value is set for weight threshold of 300 pounds, and this field has been set to 3 days, the shipment is shipped after 3 days, regardless of whether the weight threshold has been met.</p>
Consolidate up to weight threshold of	Enter a weight.
Consolidate up to volume threshold of	Enter a volume
Region Schema For Routing	<p>Select the applicable region schema for routing.</p> <p>Select 🌐. The Region Schema Details window appears. For more information about modifying the region schema details, refer to the <i>Yantra 7x Platform Configuration Guide</i>.</p>
Routing Guide	

**Table 14–2 Shipment Planning Window**

Field	Description
Not Maintained	Select this to use manual routing. Shipments are managed in the shipment console, and any routing guides are not consulted.
Maintained in Yantra	<p>Select this to use the Routing Guides maintained in Yantra 7x to determine how shipments should be routed. See <a href="#">Section 14.1.3, "Creating a Routing Guide"</a> on page 90.</p> <p>In addition to the routing guide maintained here by the enterprise, there may be a routing guide for the buyer organization. For more information about configuring routing guides for buyer organizations, see the <i>Yantra 7x Platform Configuration Guide</i>.</p> <p>See <a href="#">Section 14.1.1.1, "Creating a Freight Term"</a> on page 84 for details about using both buyer and enterprise routing guides.</p>
Maintained Externally	<p>Select this to indicate that an external routing system is used. The routing guides maintained in Yantra 7x are not consulted.</p> <p>Examples of external routing systems include using an integrated Transportation Management System (TMS), or implementing a User Exit which consults with the buyer organization.</p>

### 14.1.3 Creating a Routing Guide



*Routing Guides* are a list of conditions which determine how a shipment should be routed. A routing guide has a time period for which is effective, and conditions for when it should be applied. These conditions are based on Freight Terms and Department.

Each routing guide contains a list of *routing guide lines*, each of which describe detailed conditions for selecting a carrier. The routing guide information is based on data used by VICS (Voluntary InterIndustry Commerce Standards) routing.

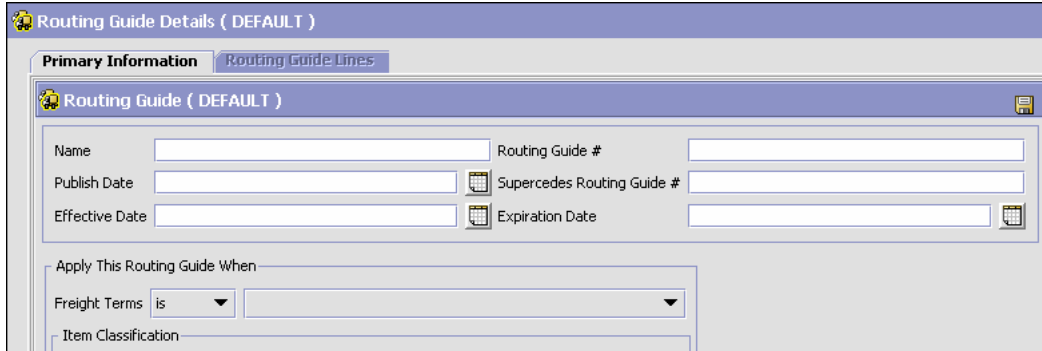
To create a routing guide:

1. From the tree in the application rules side panel, choose Logistics Management > Logistics > Shipment Planning. The Shipment Planning window appears in the work area.



2. Select  on the Routing Guides list window. The Routing Guide Details window appears in the work area.
3. Choose .

**Figure 14–2 Routing Guide Details Window**



**Table 14–3 Routing Guide Details Window**



Field	Descriptions
Name	Enter a name for the routing guide.
Routing Guide #	A number for the routing guide.
Publish Date	When this routing guide is available within the system.
Supersedes Routing Guide #	Tracking information. For example, if a minor revision is made to routing guide "1234", you might create a routing guide "1234-A", and enter that it supersedes routing guide "1234". This is for information purposes only.
Effective Date	The start date for applying the routing information in this routing guide. You can use the effective date and expiration date to apply routing guidelines for particular periods of time.
Expiration Date	The end date for applying the routing information in this routing guide.
Apply this Routing Guide when	

**Table 14–3 Routing Guide Details Window**

Field	Descriptions
Freight Terms	<p>Apply this routing guide when this condition is met. Select <i>is</i>, <i>is in</i>, or <i>is not in</i>. Use:</p> <ul style="list-style-type: none"> <li>• <i>is</i> to specify a single Freight Term.</li> <li>• <i>is in</i> to specify a group of Freight Terms, one of which must be matched.</li> <li>• <i>is not in</i> to specify a group of Freight Terms. The routing guide is used if the Freight Term does not match one of these values.</li> </ul>
Item Classification	<p>Items can be classified.</p> <p><b>Note:</b> This field appears when valid item classifications have been set up for Routing Guide.</p>

## 14.1.4 Modifying a Routing Guide

To modify a routing guide:

1. From the tree in the application rules side panel, choose Logistics Management > Shipment Planning. The Shipment Planning window appears in the work area.
2. Select a routing guide in the Routing Guide list window, and select .
3. The Routing Guide Details window appears in the work area.
4. Enter information in the applicable fields. Refer to [Table 14–3](#) for field value descriptions.
5. Choose .

### 14.1.4.1 Creating a Routing Guide Line

*Routing guide lines* contain the specific conditions to use when routing a shipment. A routing guide can contain multiple routing guide lines.

When routing occurs, the shipment is matched against the routing guide lines. Based on the criteria specified, a carrier and carrier service is selected. The shipment mode is determined for shipments/loads based on the combination of the carrier and carrier service.



When routing results in a change to the shipment destination, system will route once again, with the revised destination as the factor for routing. This type of configuration is used for consolidator nodes. While routing the second time, system looks for the routing guide entry that contains destination node, but without any other destination parameters filled out (such as address, country, etc). However, the consolidator destination node should be defined.

To create a routing guideline:


1. From the Routing Guidelines Details window, select the Routing Details Tab. To have access to the Routing Details Tab, save the information you have entered on the Primary Info Tab.
2. A Routing Guide Line search window appears.

**Figure 14–3 Routing Guide Line Window**

The screenshot shows a software window titled "Routing Guide Details ( DEFAULT )". It has two tabs: "Primary Info" and "Routing Guide Lines", with the latter being selected. Below the tabs is a "Routing Guide Line Search" section containing several input fields: "Name" (with the value "Routing Guide 1"), "Routing Guide # 1", "Carrier/Service" (a dropdown menu), and "Max Records" (with the value "1,000"). There are also icons for a folder and a refresh button. Below the search section is a "Search Results" section with a table of columns: "Ship From Node", "Ship From State", "Ship From City", "Ship From Post...", "Ship To Node", "Ship To Region", "Ship To State", "Ship To City", and "Ship To Postal ...". The table is currently empty, and the status at the bottom left reads "Results 0 Of 0".

3. Select . A Routing Guide Line Details screen appears in the work area.
4. Enter information in the applicable fields. Refer to [Table 14–4](#) for field value descriptions.
5. Choose .

**Figure 14–4 Routing Guide Line Details Window**


**Routing Guide Line Details**

When shipping from:


☒ Node
 is

☐ When ship from is not node, select the following attribute(s)

Country is  State is

City is  Zip Code is

And shipping to:

☒ Node
 is
 
☐ Region
 


☐ When ship to is not node and region, select the following attribute(s)

Country is  State is

City is  Zip Code is

Consolidator is  Store # is

And weight is in the range:

From  LBS

To  LBS

And volume is in the range:

From  CIN


To  CIN

And handling units are in the range:

From




To

And if requested carrier service code is

Carrier Service Code  

Then ship via:

**Carrier Services**


Priority	Carrier/Service	Break Bulk Node	Contact Specified

Results 0 Of 0

With overrides:

☒ Override Freight Terms
  COLLECT

☐ Override Ship To


**Routing Guide Line Details**

**Table 14–4 Routing Guide Line Details**

Setting conditions	
In many of the following fields, you can select is, is in, or is not in, and then specify a value. Use:	
<ul style="list-style-type: none"> <li>• <i>is</i> to specify that a single value must be matched</li> <li>• <i>is in</i> to specify a group of values, one of which must be matched.</li> <li>• <i>is not in</i> to specify a group of values. The routing guide line is used if none of these values match.</li> </ul>	
For example to match any one of a group of states, specify	
State <i>is in</i> California, Washington, Oregon, Nevada.	
When assessing the condition, California would match, Florida would not.	
Field	Description
When shipping from:	
Node	Select the node.
When ship from is not node, select the following attribute(s)	Select one or more of the following conditions.
Country	Select the country name(s).
State	Enter the state name(s).
City	Enter the city name(s).
Zip Code	Enter the zip code or zip code range.
And shipping to:	
Node	Select the node.
Region	Select the region.
When ship to is not node and region, select the following attribute(s)	Select one or more of the following conditions.
Country	Select the country name(s).
State	Enter the state name(s).
City	Enter the city name(s).
Zip Code	Enter the zip code or zip code range.

**Table 14–4 Routing Guide Line Details**

Consolidator	Select the consolidator name(s)
Store#	Select the store number(s).
And weight is in the range:	You can match weight. For example, if you want packages that weigh between 100 and 500 pounds to be shipped using a specific carrier, you would specify From as '100' and To as '500'.
From	Enter the minimum value.
To	Enter the maximum value.
And volume is in the range:	You can match volume. For example, if you want packages that are between 3 and 10 cubic feet to be shipped using a specific carrier, you would specify From as '3' and To as '10'.
From	Enter the minimum value.
To	Enter the maximum value.
And handling units are in the range:	Number of cases.
From	Enter the minimum value.
To	Enter the maximum value.
And if requested carrier service code is	
Carrier Service Code	Select a carrier service code.
For details about defining carrier services, see <a href="#">Section 14.1.4.1.1, "Defining Carrier Services"</a> on page 98.	
Then ship via:	
Priority	Indicates the number to give this rule a relative importance.  When a shipment is compared to the routing guide lines, there may be two carrier services that could be used. The priority serves as a tie breaker. The carrier service with the lowest number is used.
Carrier / Service	Indicates the carrier and service code that is desired.
Break Bulk Node	The break bulk location that is close to the buyer.
Contact Specified	Indicates whether the contact details for the carrier is specified.

Table 14–4 Routing Guide Line Details

With overrides:							
Override Freight Terms	Select to override the shipment's Freight Term.						
Override Ship To	<div>To override the Ship To value, check this box and select one of the following:<table><tr><td>Node</td><td>Select the node name.</td></tr><tr><td>Consolidator</td><td>Select the consolidator name.</td></tr><tr><td>Store#</td><td>Select the store number.</td></tr></table><div>This is only used when performing routing again due to a revised ship to address.</div></div>	Node	Select the node name.	Consolidator	Select the consolidator name.	Store#	Select the store number.
Node	Select the node name.						
Consolidator	Select the consolidator name.						
Store#	Select the store number.						

When the conditions set are assessed, the routing guide line that matches most conditions is used. For example, imagine there are three routing guide lines:

- Routing guide line A* - What to do when shipping from Massachusetts
- Routing guide line B* - What to do when shipping from Massachusetts, and when shipping from the zip code 01810.
- Routing guide line C* - What to do when shipping from Massachusetts or NY.

If the shipment originates from the zip code 01810, it matches all of these routing guide lines. The actions specified in *Routing guide line B* is used, as more conditions are met (both the state and the zip code).

If the shipment originates from Massachusetts, but not from zip code 01810, then both *Routing guide line A* and *Routing guide line C* match.

14.1.4.1.1 Defining Carrier Services

When routing occurs, the shipment is matched against the routing guide lines. Based on the criteria specified, you can select a carrier service to use. For example, to determine a carrier service, the following conditions are assessed:

- Whether the carrier is capable of shipping all items in the shipment.



- If multiple carriers are determined for the shipment, then the shipping costs is assessed. The carrier with lower shipping costs is being considered.
- If the cost is the same for both the carriers, then the carrier with the lowest numbered priority is being selected.

You can use the Carrier Services panel for:

- [Creating a Carrier Service](#)
- [Modifying a Carrier Service](#)
- [Deleting a Carrier Service](#)


## Creating a Carrier Service

To create a carrier service:


1. From the Routing Guidelines Details window, in the Carrier Services panel, select . The Carrier Services window appears.



The screenshot shows the 'Carrier Services' window. It has a title bar with the text 'Carrier Services' and a save icon. The main area contains three input fields: 'Priority' (a text box), 'Carrier/Service' (a dropdown menu with a small icon to its right), and 'Break Bulk Node' (a dropdown menu). Below these fields is a section titled 'Contact Address' with a small icon to its left and a gear icon to its right. This section contains a large text area for input. At the bottom of the window, there are two tabs: 'Address' (with a small icon) and 'Contact Info' (with a small icon).



2. Enter information in the applicable fields. Refer to [Table 14–5](#) for field value descriptions.
3. Choose .

**Table 14–5 Carrier Services**

Fields	Description
Priority	Enter a number to give this rule a relative importance. When a shipment is compared to the routing guide lines, there may be two carrier services that could be used. The priority serves as a tie breaker. The carrier service with the lowest number is used.
Carrier/Service	Select the carrier or service code that is desired.
Break Bulk Node	Select the break bulk location.
Contact Address	Used to specify the contact address information. Click  to change the contact Address.



## Modifying a Carrier Service

To modify a carrier service:

1. From the Routing Guidelines Details window, in the Carrier Services panel, select a carrier service in the Carrier Services list window, and select . The Carrier Services window appears.
2. Enter information in the applicable fields. Refer to [Table 14–5](#) for field value descriptions.
3. Choose .

## Deleting a Carrier Service



To modify a carrier service:

1. From the Routing Guidelines Details window, in the Carrier Services panel, select a carrier service in the Carrier Services list window and select . The Carrier Services window appears.
2. Choose .

### 14.1.4.2 Modifying a Routing Guide Line


To modify a routing guide line:

1. From the Routing Guidelines Details window, select the Routing Details Tab. A Routing Guide Line search window appears.

2. Select a routing guide line in the Routing Guide Line list window and select . The Routing Guide Line Details window appears.
3. Enter information in the applicable fields. Refer to [Table 14–4](#) for field value descriptions.
4. Choose .


#### 14.1.4.3 Deleting a Routing Guide Line

To delete a Routing Guide Line:

1. From the Routing Guide Lines Details window, select the Routing Details Tab. A Routing Guide Line search window appears.
2. Select a routing guide line in the Routing Guide Line list window and choose .

### 14.1.5 Deleting a Routing Guide

To delete a routing guide:

1. From the tree in the application rules side panel, choose Logistics Management > Shipment Planning. The Shipment Planning window appears in the work area.
2. Select the applicable Routing Guide and choose .



## Time-Triggered Transaction Reference

---

Yantra 7x provides a collection of time-triggered transactions, which are utilities that perform a variety of individual functions, automatically and at specific time intervals.

Time-triggered transactions perform repetitive actions on a scheduled basis, typically performing database updates, raising events, or calling APIs. One type of transaction, monitors, are designed to watch for processes or circumstances that are out of bounds and then raise alerts. Often, but not always, they retrieve tasks from the task queue or work from the pipeline.

Some transactions enable you to collect statistical data regarding the application's health. This data is collected periodically, using the value specified for the `yantra.statistics.persist.interval` attribute in the `yfs.properties` file. By default, statistics collection set to "on."

For more information about statistics persistence, see the *Yantra 7x Performance Management Guide*. For more information about the specific statistics parameters used, see the applicable time-triggered transactions.

The time-triggered transactions described in this appendix are unique transactions, that may or may not be document type specific. For document specific transactions, the nomenclature helps define which unique transaction it is based on: a transaction ID will be in the format `Unique_Transaction_ID.Document_Type_Code`. For example, the transaction ID for Purge Return is `PURGE.0003`, indicating that it is based on the unique transaction `PURGE`, for document type `0003`, which is Return Order. Therefore, in order to be able to configure Purge Return, you should look for the `PURGE` transaction ID in this appendix, which is Order Purge.

Yantra 7x provides the following types of time-triggered transactions:

- [Business Process Time-Triggered Transactions](#) - responsible for processing
- [Time-Triggered Purge Transactions](#) - clear out data that may be discarded after having been processed
- [Task Queue Syncher Time-Triggered Transactions](#) - update the task queue repository with the latest list of open tasks to be performed by each transaction, based on the latest pipeline configuration.
- [Monitors](#) - watch and send alerts for processing delays and exceptions

Yantra 7x tracks the following statistics for each time-triggered transaction:

- `ExecuteMessageCreated` - The number of jobs added to the JMS queue in a given time interval.
- `ExecuteMessageSuccess` - The number of jobs that were executed successfully in a given time interval.
- `ExecuteMessageError` - The number of jobs that failed to execute in a given time interval.
- `GetJobsProcessed` - The number of `GetJob` messages that were processed in a given time interval.

---

---

**Note:** Some of the statistics collected and tracked in Release 7.5 SP1 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Yantra.

---

---

## A.1 Running Time-Triggered Transactions

All time-triggered transactions are threadable. This means that you can run multiple instances of a transaction within a single process. For information on running time-triggered transactions, see the *Yantra 7x Installation Guide*. For information on fine-tuning system performance while running them concurrently, see the *Yantra 7x Performance Management Guide*.

## A.2 Business Process Time-Triggered Transactions

This section provides an alphabetical list of all business process transactions.

---

---

**Note:** Some of the statistics collected and tracked in Release 7.5 SP1 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Yantra.

---

---



---

---

**Note:** All Business Process Time-Triggered Transactions have a `CollectPendingJobs` criteria parameter. If this parameter is set to `N`, the agent will not collect information on the pending jobs for that time-triggered transaction. This pending job information is used for monitoring the agent in the *Yantra 7x System Management Guide*.

By default, `CollectPendingJobs` is set to `Y`. It can be helpful to set it to `N` if one particular time-triggered transaction is performing a significant amount of `getPendingJobs` queries, and the overhead cost is too high.

---

---

### A.2.1 Change Load Status

This transaction is equivalent to the `changeLoadStatus()` API. For detailed information about this transaction, see *Yantra 7x Javadocs*.

To be configured as part of your load processing pipeline, this transaction can be used whenever an automatic change in the status of a load is required. This automatic change could represent exporting load information to load planning software or transmission to the load's carrier.

**Note:** This transaction should be configured to work from the task queue.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–1** *Change Load Status Attributes*

Attribute	Value
Base Transaction ID	CHANGE_LOAD_STATUS
Base Document Type	Load
Base Process Type	Load Execution
Abstract Transaction	Yes
APIs Called	changeLoadStatus ( )

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–2** *Change Load Status Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–3** *Change Load Status Statistics*

Statistic Name	Description
NumLoadsChanged	Number of loads whose status was changed.



### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the `AVAILABLE_DATE` value less than or equal to (`<=`) the current date value in the `YFS_Task_Q` table.

### Events Raised

This transaction raises events specified by the `changeLoadStatus()` API in the *Yantra 7x Javadocs*.

## A.2.2 Change Shipment Status

This transaction is equivalent to the `changeShipmentStatus()` API. For detailed information about this transaction, see *Yantra 7x Javadocs*.

To be configured as part of your shipment processing pipeline, this transaction can be used whenever an automatic change in the status of a shipment is required. For example, this automatic change could represent exporting shipment information to a warehouse management system or to transmit an Advance Shipping Notice to the buyer.

**Note:** This transaction should be configured to work from the task queue.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–4** *Change Shipment Status Attributes*

Attribute	Value
Base Transaction ID	CHANGE_SHIPMENT_STATUS
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	Yes
APIs Called	None

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–5** *Change Shipment Status Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–6** *Create Chained Order Statistics*

Statistic Name	Description
NumShipmentsChanged	Number of shipments whose status was changed.

## Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the `AVAILABLE_DATE` value less than or equal to (`<=`) the current date value in the `YFS_Task_Q` table.

## Events Raised

This transaction raises events specified by the `changeShipmentStatus()` API in the *Yantra 7x Javadocs*.

## A.2.3 Close Delivery Plan

To boost system performance, this transaction serves as a temporary purge until the Delivery Plan Purge deletes delivery plan-related data (see [Section A.3.3.3, "Delivery Plan Purge"](#) on page 181).

This transaction picks all delivery plans that do not have any of their loads or shipments still open and marks the `deliveryplan_closed_flag='Y'`. This flag indicates no further operations are possible on the plan.

This transaction corresponds to the base transaction close delivery plan (CLOSE\_DELIVERY\_PLAN) in the load pipeline.

Any enterprise using the Yantra 7x Application Consoles must schedule purge jobs.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–7 Close Delivery Plan Attributes**

Attribute	Value
Base Transaction ID	CLOSE_DELIVERY_PLAN
Base Document Type	Load
Base Process Type	Load Execution
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–8 Close Delivery Plan Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–9 Close Delivery Plan Statistics**

Statistic Name	Description
NumDeliveryPlansClosed	Number of delivery plans closed.

Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to (<=) the current date value in the YFS\_Task\_Q table.

Events Raised

The following events are raised by this time-triggered transaction:

Table A–10 Events Raised by Close Delivery Plan Transaction

Transaction/Event	Key Data	Data Published	Template Support?
ON_SUCCESS	delivery_plan_dbd.txt	YDM_CLOSE_DELIVERY_PLAN.ON_SUCCESS.xml	Yes

However, note that the template name would read <TransactionId>.ON\_SUCCESS.xml.

A.2.4 Close Load

To boost system performance, this transaction serves as a temporary purge until the Load Purge deletes load-related data (see [Section A.3.3.9, "Load Purge"](#) on page 193).

This transaction corresponds to the base transaction Close Load (CLOSE\_LOAD) in the load pipeline.

If you use the Load processing pipeline, you must schedule this transaction. Only closed loads are picked up by the purge transaction. Therefore, it is required that this transaction be made part of the pipeline and scheduled to run at the end of the day.

**Note:** This transaction should be made part of the pipeline. In addition, it should be configured to work from the task queue.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–11 Close Load Attributes**

Attribute	Value
Base Transaction ID	CLOSE_LOAD
Base Document Type	Load
Base Process Type	Load Execution
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–12 Close Load Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:

Table A–13 Close Load Statistics

Statistic Name	Description
NumLoadsClosed	Number of loads closed.

Pending Job Count

For this transaction the pending job count is the number of open delivery plans, which are not associated to any open loads and open shipments.

Events Raised

The following events are raised by this time-triggered transaction:

Table A–14 Events Raised by the Close Load Transaction

Transaction/Event	Data Published	Template Support?
ON_SUCCESS	YDM_CLOSE_LOAD_PLAN.ON_SUCCESS.xml	Yes

However, note that the template name would read <TransactionId>.ON\_SUCCESS.xml.

A.2.5 Close Manifest

This time-triggered transaction sets the manifest’s MANIFEST\_CLOSED\_FLAG flag to ‘Y’ and updates the manifest status to CLOSED. This time-triggered transaction confirms all the shipments that are pending confirmation, and closes the manifest.

**Note:** If the Close Manifest Agent is triggered without any criteria it closes all the candidate manifests across all ShipNodes.

The `yfs.closemanifest.online` property in the `yfs.properties` file is used to set this time-triggered transaction to work in online or offline mode.

- **Online mode:** In the online mode, the close manifest transaction runs as usual, confirming all shipments in the manifest and then closing the manifest.
- **Offline mode:** In the offline mode, the close manifest transaction triggers an agent and changes the manifest status to 'Closure Requested'. When the agent runs, it confirms either each shipment of the manifest, or closes the manifest, in an execution call.

The mode of operation (online or offline) is decided on the basis of a property defined in the `yfs.properties` file:

```
yfs.closemanifest.online = Y/N
```

The default out-of-the-box shipped property causes the Close Manifest transaction to run in online mode.

**Note:** In instances where the Close Manifest transaction is run in offline mode, ensure that all Agent Criteria defined for the transaction are configured properly.

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–15** *Close Manifest Attributes*

Attribute	Value
Base Transaction ID	CLOSE_MANIFEST
Base Document Type	General
Base Process Type	Manifesting
Abstract Transaction	No
APIs Called	confirmShipment()

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–16** *Close Manifest Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that will only perform their tasks on the nodes with a matching node transactional velocity value.  Valid values are: LOW, HIGH, and any additional values defined by the Hub from Platform > System Administration > Agent Criteria Groups.
ShipNode	Optional. Ship node for which the Close Manifest needs to be run. If not passed, then all ship nodes are monitored.

### Statistics Tracked

The following are statistics are tracked for this transaction:

**Table A–17** *Close Manifest Statistics*

Statistic Name	Description
NumShipmentsConfirmed	Number of shipments confirmed.
NumManifestsClosed	Number of manifests closed.
NumManifestsErrored	Number of manifests errored.
NumShipmentsErrored	Number of shipments errored.

### Pending Job Count

For this transaction the pending job count is the sum of open manifests and shipments belonging to manifests (with MANIFEST\_STATUS='1200').

### Events Raised

The following events are raised by this time-triggered transaction:



*Table A–18 Events Raised by the Close Manifest Transaction*

Transaction/Event	Key Data	Data Published	Template Support?
ON_SUCCESS	manifest_dbd.txt	YDM_CLOSE_MANIFEST.ON_SUCCESS.xml	Yes

## A.2.6 Close Order

This time-triggered transaction sets the order's ORDER\_CLOSED flag to 'Y' and raises the ON\_SUCCESS event. These actions are only performed when the entire ORDER\_QTY all the order lines reach the configured pickup status(es). If an order has ORDER\_CLOSED set to 'Y', it is not picked up for monitoring.

**Note:** The Close Order agent must be configured along with the Purge transaction in the pipeline.

**Note:** Many of this transaction's elements and attributes are template driven. Refer to the XML for element level details.

### Attributes

The following are the attributes for this time-triggered transaction:

*Table A–19 Close Order Attributes*

Attribute	Value
Base Transaction ID	CLOSE_ORDER
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–20 Close Order Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–21 Close Order Statistics**

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumOrdersClosed	Number of orders closed.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table, if tasks on hold are not ready to be processed.

### Events Raised

The following events are raised by this time-triggered transaction:

**Table A–22 Events Raised by the Close Order Transaction**

Transaction/Event	Data Published	Template Support?
ON_SUCCESS	YFS_CLOSE_ORDER.ON_SUCCESS.xml	Yes

## A.2.7 Close Receipts

This time-triggered transaction closes receipts using the receiving rule specified.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–23 Close Receipts Attributes**

Attribute	Value
Base Transaction ID	RECEIPT_COMPLETE
Base Document Type	Order
Base Process Type	Receipt (Purchase Order Receipt, Return Receipt, Transfer Order Receipt)
Abstract Transaction	No
APIs Called	None
User Exits Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–24 Close Receipts Criteria Parameters**

Parameter	Description
Action	Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Enterprise for which the Close Receipts needs to be run. If not passed, then all enterprises are monitored.

**Table A–24 Close Receipts Criteria Parameters**

Parameter	Description
Node	Node for which the Close Receipts Purge needs to be run. If not passed, then all nodes are monitored.
AgentCriteriaGroup	Used to classify nodes. This value can be accepted by WMS time-triggered transactions that will only perform their tasks on the nodes with a matching node transactional velocity value.  Valid values are: LOW, HIGH, and any additional values defined by the Hub from Platform > System Administration > Agent Criteria Groups.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–25 Close Receipts Statistics**

Statistic Name	Description
NumReceiptsClosed	Number of receipts closed.

### Pending Job Count

For this transaction the pending job count is the number of Receipts that can be closed (with OPEN\_RECEIPT\_FLAG='Y').

### Events Raised

The following events are raised by this time-triggered transaction:

**Table A–26 Events Raised by the Close Receipts Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
ON_SUCCESS	receipt_dbd.txt	YFS_RECEIPT_COMPLETE.ON_SUCCESS.xml	Yes

---

**Troubleshooting Tip:** When multiple inbound shipments are received into the same location, and the inventory received is not license plated, an error message, "There is no inventory for put away at the SourceLocation" appears. The solution to this problem lies in one of these steps:

- Manually create move requests for receipts that you already received. For more information about creating move requests, refer to the *Yantra 7x Warehouse Management System User Guide*.
  - For receipts that are expected to be received, ensure that the inventory is license plated and that you don't receive inbound shipments and inventory for put away into the same location.
- 

## A.2.8 Close Shipment

To boost system performance, this transaction serves as a temporary purge until the Shipment Purge deletes all shipment-related data (see [Section A.3.3.23, "Shipment Purge"](#) on page 228).

This transaction picks all shipments eligible to be closed, based on the pipeline configuration for pickup for transaction CLOSE\_SHIPMENT, and marks the shipment\_closed\_flag='Y'. This flag indicates no further operations are possible on the shipment. There is no status change involved. This transaction can be configured in the pipeline so that it picks up either Shipped or Delivered status.

This transaction corresponds to the base transaction close shipment (CLOSE\_SHIPMENT) in the shipment pipeline.

---

**Note:** This transaction should be made part of the pipeline. In addition, it should be configured to work from the task queue.

---

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–27 Close Shipment Attributes**

Attribute	Value
Base Transaction ID	CLOSE_SHIPMENT
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–28 Close Shipment Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following are statistics are tracked for this transaction:

**Table A–29 Close Shipment Statistics**

Statistic Name	Description
NumShipmentsClosed	Number of shipments closed.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE

value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

### Events Raised

The following events are raised by this time-triggered transaction:

**Table A–30 Events Raised by the Close Shipment Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
ON_SUCCESS	shipment_dbd.txt	YDM_CLOSE_SHIPMENT.ON_SUCCESS.xml	Yes

## A.2.9 Collect Shipment Statistics

Collect Shipment Statistics is a time-triggered transaction which can be invoked to process the shipments, and generate information required for the Daily Shipment Report.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–31 Collect Shipment Statistics Attributes**

Attribute	Value
Transaction Name	Collect Shipment Statistics
Transaction ID	COLLECT_STATISTICS
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	No
APIs Called	None
User Exits Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–32 Collect Shipment Statistics Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Node	Required. The warehouse management ship node for which records are being processed.
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that will only perform their tasks on the nodes with a matching node transactional velocity value.  Valid values are: LOW, HIGH, and any additional values defined by the Hub from Platform > System Administration > Agent Criteria Groups.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–33 Statistics for Collect Shipment Statistics**

Statistic Name	Description
NumDaysStatisticsCollected	Number of days for which shipment statistics have been collected.

### Pending Job Count

For this transaction the pending job count is the number of days for which shipment statistics needs to be collected. The number of days is calculated as the difference (in days) between the current date and the last date when shipment statistics was collected.

### Events Raised

The following events are raised by this time-triggered transaction:



**Table A–34 Events Raised by the Collect Shipment Statistics Transaction**

Transaction/Event	Data Published	Template Support?
ON_SUCCESS	YDM_COLLECT_STATISTICS.ON_SUCCESS.xml	No

## A.2.10 Complete Planned Order

Complete Planned Order takes planned orders to completion after negotiations are resolved. Use this time-triggered transaction on a planned order after negotiation is complete. This time-triggered transaction is being deprecated for Version 5.0-SP1.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–35 Complete Planned Order Attributes**

Attribute	Value
Transaction Name	Complete Planned Order
Transaction ID	PLAN_ORDER_COMPLETE
Base Document Type	Order
Base Process Type	Planned Order Execution
Abstract Transaction	No
APIs Called	None
User Exits Called	None

### Criteria Parameters

The following are the parameters for this transaction:

**Table A–36 Order Complete Criteria Parameters**

Parameter	Description
DocumentType	Required. The type of document to process for a particular run. Valid values are: <ul style="list-style-type: none"> <li>• 0001 - Sales Order (Default)</li> <li>• 0002 - Planned Order</li> </ul>
TotalRecords	Optional. Number of records for the time-triggered transaction to pass. If not passed, defaults to 5000.

### Statistics Tracked

None.

### Pending Job Count

None.

### Events Raised

The following events are raised by this time-triggered transaction:

**Table A–37 Events Raised by the Order Complete Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
PLAN_ORDER_COMPLETE	modifyOrder_dbd.txt	YFS_getPlannedOrderStatus_Output.xml	No

## A.2.11 Consolidate Additional Inventory

The Consolidate Additional Inventory time-triggered transaction consolidates supply and demand from the YFS\_INVENTORY\_SUPPLY\_ADDNL and YFS\_INVENTORY\_DEMAND\_ADDNL tables. Consolidation is performed by summing up the quantities of additional supply and demand in the YFS\_INVENTORY\_SUPPLY and YFS\_INVENTORY\_DEMAND tables.

If no matching supply or demand is found, a new supply or demand is created with the sum quantity of the changes in the YFS\_INVENTORY\_SUPPLY\_ADDNL and YFS\_INVENTORY\_DEMAND\_ADDNL tables. After the changes are applied, the records in the YFS\_INVENTORY\_SUPPLY\_ADDNL and YFS\_INVENTORY\_DEMAND\_ADDNL tables that were used in the consolidation process, are deleted.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–38 Consolidate Additional Inventory Attributes**

Attribute	Value
Base Transaction ID	CONSOLIDATE_ADDNL_INV
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the parameters for this transaction:

**Table A–39 Consolidate Additional Inventory Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.r
Number of Records To Buffer	Optional. Number of inventory item records (whose additional supplies and demands will be consolidated_ to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–40 Consolidate Additional Inventory Statistics**

Statistic Name	Description
NumInventorySupplyAddnlsProcessed	Number of additional inventory supply records processed in the consolidation.
NumInventoryDemandAddnlsProcessed	Number of additional inventory demand records processed in the consolidation.
NumInventoryDemandDtlsProcessed	Number of inventory demand details records processed in the consolidation.

**Pending Job Count**

For this transaction the pending job count is the number of distinct inventory items in the YFS\_Inventory\_Supply\_Addnl and YFS\_Inventory\_Demand\_Addnl tables, multiplied by two.

**Events Raised**

None.

**A.2.12 Consolidate To Shipment**

This is a task queue based transaction in the order pipeline that corresponds to base transaction CONSOLIDATE\_TO\_SHIPMENT. This transaction finds a shipment into which a given order release can be included. If it finds an existing shipment, it calls `changeShipment()` API. Otherwise, it calls the `createShipment()` API.

To find the existing shipments it matches ShipNode, ShipTo Address, SellerOrganizationCode, Carrier, DocumentType and so forth, of the Order Release with that of existing shipments. List of attributes it matches is actually based on Document Template for Document Type of the Order.

This transaction is applicable only to the shipments in one of the following Statuses:

- Shipment Created
- ESP Check Required

- On ESP Hold
- Released from ESP Hold
- Released For Routing
- Awaiting Routing
- Shipment Routing
- Sent To Node
- Shipment Being Picked

**Troubleshooting Tip:** To successfully consolidate an Order Release to an existing shipment, the Add Line and related modification types on shipment in its current status should be allowed.

For more details, see the `createShipment()`, `changeShipment()`, and `releaseOrder()` APIs in the *Yantra 7x Javadocs*.

**Note:** This transaction is a part of the Order Fulfillment pipeline. In addition, it should be configured to work from the task queue.

**Note:** Order releases with GIFT\_FLAG set to `Y` will never be consolidated with any other release.

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–41 Consolidate to Shipment Attributes**

Attribute	Value
Base Transaction ID	CONSOLIDATE_TO_SHIPMENT
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No

**Table A–41 Consolidate to Shipment Attributes**

Attribute	Value
APIs Called	<code>createShipment()</code> and <code>changeShipment()</code>
User Exits	<ul style="list-style-type: none"> <li>It calls <code>beforeConsolidateToShipment</code> in <code>com.yantra.ydm.japi.ue.YDMBeforeConsolidateToShipment</code> for each release before it begins processing.</li> <li>After it finds the shipments, it calls <code>determineShipmentToConsolidateWith</code> in <code>com.yantra.ydm.japi.ue.YDMDetermineShipmentToConsolidateWith</code>. See <i>Yantra 7x Javadocs</i>.</li> </ul>

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–42 Consolidate to Shipment Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to <code>Get</code> , the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:

## Pending Job Count

**Table A–43 Consolidate to Shipment Statistics**

Statistic Name	Description
NumOrderReleasesConsolidated	Number of order releases consolidated.

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

## Events Raised

The following events are raised by this time-triggered transaction:

**Table A–44 Events Raised by the Consolidate to Shipment Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
ON_SUCCESS	shipment_dbd.txt	YDM_CONSOLIDATE_TO_SHIPMENT.ON_SUCCESS.xml	Yes

**Note:** This transaction also raises events specified by the `createShipment()` or `changeShipment()` APIs in the *Yantra 7x Javadocs*.

However, note that the template name would read `<TransactionId>.ON_SUCCESS.xml`.

## A.2.13 Create Chained Order

This transaction creates one or more chained orders from an order whose OrderHeaderKey is stored in the task queue object. Chainable lines of the order can also be added to existing chained orders, instead of creating new chained orders with these lines. The existing chained orders must be identified by the `determineChainedOrderForConsolidation` user exit. If the user exit is not implemented, or if the user exit returns a blank document, one or more new chained orders are created.

For more information on creation of chained orders, see the documentation on the `createChainedOrder()` API and the `YFSDetermineChainedOrderForConsolidation` user exit in the *Yantra 7x Javadocs*.

This transaction should be invoked after order scheduling.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–45 Create Chained Order Attributes**

Attribute	Value
Base Transaction ID	CHAINED_ORDER_CREATE
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	Yes
APIs Called	<code>createChainedOrder()</code>

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–46 Create Chained Order Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:



**Note:** If there are 2 orders being processed and the first order creates a  
**Table A–47 Create Chained Order Statistics**

Statistic Name	Description
NumOrdersProcessed	Number of orders processed for creating chained order.
NumOrdersCreated	Number of chained orders created.

chained order, the DetermineChainedOrderForConsolidation user exit causes the lines of the 2nd order to be added to the first order. The number of chained orders created is counted as 2.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

### Events Raised

This transaction raises events specified by the createChainedOrder() API in *Yantra 7x Javadocs*.

## A.2.14 Create Derived Order

This transaction creates one or more derived orders from an order whose OrderHeaderKey is stored in the task queue object. For existing derived orders, you can add derivable lines or create new derived orders with these lines. The existing derived orders must be identified by the determineDerivedOrderForConsolidation user exit. If the user exit is not implemented or if the user exit returns a null document, new derived orders are created. For more information on creation of derived orders, see the createDerivedOrder() API and YFSDetermineDerivedOrderForConsolidation user exit in *Yantra 7x Javadocs*.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–48** *Create Derived Order Attributes*

Attribute	Value
Base Transaction ID	DERIVED_ORDER_CREATE
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	Yes
APIs Called	<code>createDerivedOrder()</code>

**Note:** The TransactionKey posted in the task queue object must be an instance of the Abstract Transaction DERIVED\_ORDER\_CREATE for the ProcessType associated with the Order. Otherwise, an exception is thrown.

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–49** *Create Derived Order Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–50** *Create Derived Order Statistics*

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumOrdersCreated	Number of derived orders created.

**Note:** If there are 2 orders being processed and the first order creates a derived order, the DetermineChainedOrderForConsolidation user exit causes the lines of the 2nd order to be added to the first order. The number of derived orders created is counted as 2.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

### Events Raised

This transaction raises events specified by the `createDerivedOrder()` API in the *Yantra 7x Javadocs*.

## A.2.15 Create Order Invoice

This transaction creates one or more invoices from an order whose OrderHeaderKey is stored in a task queue object. The `createOrderInvoice()` API is called for the OrderHeaderKey.

Configure this transaction in the pipeline only after all processing that can impact quantity or price has been completed. Post invoice creation, the line quantity cannot be reduced below the invoiced quantity.

**Note:** Both the Create Order Invoice and Create Shipment Invoice transactions can create invoices for an Order. When configuring your pipeline, ensure that only *one* of these two transactions is configured to create invoices for a particular order line. For more information, see [Section A.2.16, "Create Shipment Invoice"](#) on page 135.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–51 Create Order Invoice Attributes

Attribute	Value
Base Transaction ID	CREATE_ORDER_INVOICE
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	Yes
APIs Called	createOrderInvoice()

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–52 Create Order Invoice Criteria Parameters

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–53 Create Order Invoice Statistics**

Statistic Name	Description
NumOrderInvoicesCreated	Number of order invoices created.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the `AVAILABLE_DATE` value less than or equal to (`<=`) the current date value in the `YFS_Task_Q` table.

### Events Raised

This transaction raises events specified by the `createOrderInvoice()` API in the *Yantra 7x Javadocs*.

## A.2.16 Create Shipment Invoice

Invoicing is mandatory if an order requires payment processing. Invoicing occurs if the following conditions are met:

- Invoicing is enabled at the document parameter level.
- The Seller requires payment processing.

This transaction creates one or more invoices for the shipment whose `ShipmentKey` is stored in the task queue object. The `createShipmentInvoice()` API is called for the `ShipmentHeaderKey`.

This transaction should be configured in the shipment pipeline only after the shipment has reached a shipped status.

**Note:** Both the Create Order Invoice and Create Shipment Invoice can create invoices for an order. When configuring your pipeline, ensure that only *one* of these two transactions is configured to create invoices for a particular order line. See [Section A.2.15, "Create Order Invoice"](#) on page 133.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–54 Create Shipment Invoice Attributes**

Attribute	Value
Base Transaction ID	CREATE_SHIPMENT_INVOICE
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	Yes
APIs Called	createShipmentInvoice()

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–55 Create Shipment Invoice Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–56 Create Shipment Invoice Statistics**

Statistic Name	Description
NumShipmentInvoicesCreated	Number of shipment invoices created.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

## Events Raised

This transaction raises events specified by the `createShipmentInvoice()` API in the *Yantra 7x Javadocs*.

## A.2.17 ESP Evaluator

The ESP Evaluator time-triggered transaction verifies whether a shipment meets certain economic shipping parameters (ESP). ESP can be configured either for buyer or enterprise, with the freight terms on the shipment determining which one is used.

If the configuration is defined to hold shipment for ESP, the shipment when created is held for ESP (with status *On ESP Hold*). This task queue based time-triggered transaction evaluates the shipment for ESP, and passes it on to the next step in the shipment pipeline if the criteria (weight and volume limits, plus maximum days of hold up) are met. The shipment status is now set to *Released from ESP hold*, and routing processing begins.

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–57 ESP Evaluator Attributes**

Attribute	Value
Base Transaction ID	ESP_EVALUATOR.0001
Base Document Type	Order
Base Process Type	Outbound Shipment
Abstract Transaction	No
APIs Called	None
User Exits Called	None

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–58** *ESP Evaluator Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
EnterpriseCode	Optional. Enterprise for which the ESP Evaluator needs to be run. If not passed, then all enterprises are monitored.
Number of Records to Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.
Node	Required. The warehouse management ship node for which records are being processed.
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that will only perform their tasks on the nodes with a matching node transactional velocity value.  Valid values are: LOW, HIGH, and any additional values defined by the Hub from Platform > System Administration > Agent Criteria Groups.

**Statistics Tracked**

None.

**Pending Job Count**

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

**Events Raised**

The following events are raised by this time-triggered transaction:



*Table A–59 Events Raised by ESP Evaluator Transaction*

Transaction/Event	Key Data	Data Published	Template Support?
ON_SUCCESS	shipment_ dbd.txt	ESP_ EVALUATOR.ON_ SUCCESS.xml	Yes

## A.2.18 Mark Load as Trailer Loaded

This is a time-triggered transaction which works on “Load pipeline”.

This time-triggered transaction gets records from the Task Q. This transaction is used to mark the load as trailer loaded when all containers for the load are on the trailer.

### Attributes

The following are the attributes for this time-triggered transaction:

*Table A–60 Mark Load As Trailer Loaded Attributes*

Attribute	Value
Base Transaction ID	MARK_AS_TRAILER_LOADED
Base Document Type	Load
Base Process Type	Load Execution
Abstract Transaction	No
APIs Called	None
User Exits Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–61 Mark Load As Trailer Loaded Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
ReprocessInterval	Optional. Reprocess Interval is the time taken to reprocess the load.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–62 Mark Load As Trailer Loaded Statistics**

Statistic Name	Description
NumLoadsChanged	Number of trailer loads changed.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

### Events Raised

None.

## A.2.19 Match Inventory

Match Inventory processes all pending records in the YFS\_INVENTORY\_SHIPMENT table. Pending records have a smaller number in POSTED\_QUANTITY than in QUANTITY.

Each pending record is matched against the receipt records in YFS\_INVENTORY\_RECEIPT table by applying the inventory cost determination logic. The unit cost at which the sales and receipt data are matched is also posted in YFS\_INVENTORY\_MATCH table.

Use this transaction if any of the configured ship nodes maintain inventory cost.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–63 Match Inventory Attributes**

Attribute	Value
Base Transaction ID	INVENTORY_MATCH
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–64 Match Inventory Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
InventoryOrganizationCode	Optional. Valid inventory owner organization. Organization to process in this run. If not passed, all inventory organizations are processed.
CutOffDate	Optional. If passed, records are matched up to this date. Defaults to all unmatched records in Database.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–65 Match Inventory Statistics**

Statistic Name	Description
NumInventoryShipmentsProcessed	Number of inventory shipments processed.
NumInventoryMatchesInserted	Number of inventory matches inserted.

**Pending Job Count**

For this transaction the pending job count is the number of distinct inventory items that exist in the YFS\_INVENTORY\_SHIPMENT table where the QUANTITY value is not equal to the POSTED\_QUANTITY value.

**Events Raised**

None.

**A.2.20 Payment Collection**

This transaction requests credit validation for orders that are pending authorization or charging.

Use this transaction for creating authorization and charge requests.

**Note:** This transaction works in combination with the Payment Execution transaction. Although this transaction can run independent of that transaction, authorization and collection occurs *only* after the Payment Execution dependencies are met. For more details, see [Section A.2.21, "Payment Execution"](#) on page 145.

**Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–66 Payment Collection Attributes for Sales Orders**

Attribute	Value
Base Transaction ID	PAYMENT_COLLECTION
Base Document Type	Order

**Table A–66** *Payment Collection Attributes for Sales Orders*

Attribute	Value
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	requestCollection()

**Table A–67** *Payment Collection Attributes for Return Orders*

Attribute	Value
Base Transaction ID	PAYMENT_COLLECTION.0003
Base Document Type	Order
Base Process Type	Reverse Logistics
Abstract Transaction	No
APIs Called	requestCollection()

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–68** *Payment Collection Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the transaction needs to be run. If not passed, then all enterprises are monitored.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–69 Payment Collection Statistics**

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumChargeReqsCreated	Number of charge requests created.
NumAuthorizationReqsCreated	Number of authorization requests created.

**Pending Job Count**

For this transaction the pending job count is the number of orders in the appropriate payment statuses with the value of the AUTHORIZATION\_EXPIRATION\_DATE is less than or equal to ( $\leq$ ) the currentdate. The appropriate payment statuses for such orders are:

- AWAIT\_PAY\_INFO
- AWAIT\_AUTH
- REQUESTED\_AUTH
- REQUEST\_CHARGE
- AUTHORIZED, INVOICED
- PAID
- RELEASE\_HOLD
- FAILED\_AUTH
- FAILED\_CHARGE
- VERIFY
- FAILED

**Events Raised**

The following events are raised by this time-triggered transaction:

**Table A–70 Events Raised by the Payment Collection Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
INCOMPLETE_PAYMENT_INFORMATION	modifyOrder_dbd.txt	YFS_PAYMENT_COLLECTION.INCOMPLETE_PAYMENT_INFORMATION.xml	Yes
PAYMENT_STATUS	YFS_PAYMENT_COLLECTION.PAYMENT_STATUS_dtd.txt	YFS_PAYMENT_COLLECTION.PAYMENT_STATUS.xml	Yes
ON_LIABILITY_TRANSFER	modifyOrder_dbd.txt	YFS_PAYMENT_COLLECTION.ON_LIABILITY_TRANSFER.xml	Yes

### A.2.21 Payment Execution

This transaction processes all requests that are pending authorization and charging.

**Note:** Use this time-triggered transaction for processing all authorization and charge requests.

This transaction requires interfacing with a product that provides financial services.

#### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–71 Payment Execution Attributes for Sales Orders**

Attribute	Value
Base Transaction ID	PAYMENT_EXECUTION
Base Document Type	Order
Base Process Type	Order Fulfillment

**Table A–71 Payment Execution Attributes for Sales Orders**

Attribute	Value
Abstract Transaction	No
APIs Called	executeCollection()
User Exits Called	collectionCreditCard, collectionOthers, collectionCustomerAcct

**Table A–72 Payment Execution Attributes for Return Orders**

Attribute	Value
Base Transaction ID	PAYMENT_EXECUTION.0003
Base Document Type	Order
Base Process Type	Reverse Logistics
Abstract Transaction	No
APIs Called	executeCollection()
User Exits Called	collectionCreditCard, collectionOthers, collectionCustomerAcct

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–73 Payment Execution Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
ChargeType	Type of credit card process. Valid values are: <ul style="list-style-type: none"> <li>AUTHORIZATION - Validates the credit card account</li> <li>CHARGE - Applies the charge to the credit card</li> </ul>



## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–74 Payment Execution Statistics**

Statistic Name	Description
NumAuthTransProcessed	Number of authorization transaction processed.
NumAuthTransSuccessfullyProcessed	Number of successful returns from user exit for authorization transaction processed.
NumChargeTransProcessed	Number of charge transaction processed.
NumChargeTransSuccessfullyProcessed	Number of successful returns from user exit for charge transaction processed.
NumCollectionValidations	Number of successful returns from the invoked validate collection user exits.
NumCreditCardCollections	Number of credit card collections.
NumCustomerAccountCollections	Number of successful returns from the customer account collection user exits.
NumOtherCollections	Number of successful returns from the other collection user exits.

## Pending Job Count

For this transaction the pending job count is the number of open charge and authorization transactions.

## Events Raised

The following events are raised by this time-triggered transaction:

**Table A–75 Events Raised by Payment Execution Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
CHARGE_FAILED	modifyOrder dbd.txt	PAYMENT_EXECUTION_ CHARGE_FAILED_ dbd.txt	No

This transaction raises events specified by the `executeCollection()` API in the *Yantra 7x Javadocs*.

## A.2.22 Post Inventory Match

This transaction processes all open records in YFS\_INVENTORY\_MATCH table and posts the records to a financial system. An open record in the YFS\_INVENTORY\_MATCH table has the status of 01. After posting, the status is changed to 02.

Use this transaction if any of the configured ship nodes maintain inventory cost.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–76 Post Inventory Match Attributes**

Attribute	Value
Base Transaction ID	POST_INVENTORY_MATCH
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–77 Post Inventory Match Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–78 Post Inventory Match Statistics**

Statistic Name	Description
NumInventoryMatchPosted	Number of inventory match records posted.

### Pending Job Count

For this transaction the pending job count is the number of inventory matches with an open status.

### Events Raised

The following events are raised by this time-triggered transaction:

**Table A–79 Events Raised by the Post Inventory Match Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
POST_INVENTORY_MATCH	POST_INVENTORY_MATCH_dbd.txt	YFS_postInventoryMatch_output.xml	No

## A.2.23 Process Order Hold Type

You can create a time-triggered transaction, derived from the abstract transaction PROCESS\_ORDER\_HOLD\_TYPE. It can be configured as the processing transaction for one or more hold types. If an order is associated with a hold type that has a transaction configured as the

processing transaction, a record is created in YFS\_TASK\_Q for processing that transaction.

When the processing transaction is triggered, it checks the hold types that it can process based on the hold type configuration. If none can be processed, the YFS\_TASK\_Q record is deleted. If some hold types can be processed, the user exit `processOrderHoldType` is invoked with the list of hold types to be processed. `processOrderHoldType` returns the list of hold types that can be removed from the order.

The transaction will modify the order and update the order hold type list based on the output of `processOrderHoldType`. If no hold types can be processed by this transaction, then the YFS\_TASK\_Q record is deleted. If some hold types still can be processed, YFS\_TASK\_Q is updated with the next available date.

The `processOrderHoldType` user exit can also be used to add hold types, and change the status of a hold type already applied to an order. For more information on the `processOrderHoldType` user exit, refer to the *Yantra 7x Javadocs*.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–80 Process Order Hold Type Attributes

Attribute	Value
Base Transaction ID	PROCESS_ORDER_HOLD_TYPE
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	Yes
APIs Called	changeOrder

Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–81 Process Order Hold Type Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

**Statistics Tracked**

None.

**Pending Job Count**

None

**Events Raised**

The following events are raised by this time-triggered transaction:

**Table A–82 Events Raised by Process Order Hold Type Transaction**

Transaction/Event	Raised when...	Key Data	Data Published	Template Support?
ON_SUCCESS	On success	modifyOrder_dbd.txt	YFS_ORDER_CHANGE.ON_SUCCESS.xml	Yes *
ON_HOLD_TYPE_STATUS_CHANGE	The status of a hold type is changed.	modifyOrder_dbd.txt	YFS_ON_HOLD_TYPE_STATUS_CHANGE.xml	Yes
* <b>Note:</b> Some of the elements and attributes are not template driven. Refer to the xml for elements level details.				

## A.2.24 Process Work Order Hold Type

This time-triggered transaction is identical to the [Process Order Hold Type](#) transaction, but it is used for work orders instead.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–83 Process Work Order Hold Type Attributes**

Attribute	Value
Base Transaction ID	PROCESS_WO_ORDER_HOLD_TYPE
Base Document Type	Work Order
Base Process Type	VAS Process
Abstract Transaction	Yes
APIs Called	modifyWorkOrder

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–84 Process Work Order Hold Type Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

None.

### Pending Job Count

None

## Events Raised

The following events are raised by this time-triggered transaction:

**Table A–85** Events Raised by Process Work Order Hold Type Transaction

Transaction/Event	Raised when...	Key Data	Data Published	Template Support?
ON_SUCCESS	On success	workOrder_dbd.txt	VAS_MODIFY_WORK_ORDER.ON_SUCCESS.xml	Yes *
ON_HOLD_TYPE_STATUS_CHANGE	The status of a hold type is changed.	workOrder_dbd.txt	VAS_ON_HOLD_TYPE_STATUS_CHANGE.xml	Yes
* <b>Note:</b> Some of the elements and attributes are not template driven. Refer to the xml for elements level details.				

## A.2.25 Publish Negotiation Results

This transaction publishes the negotiated terms to the order.

Use this transaction in environments where an order must go through a negotiation phase.

**Note:** This transaction needs to be run after negotiation is completed.

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–86** Publish Negotiation Results Attributes

Attribute	Value
Base Transaction ID	PUBLISH_ORD_NEGOTIATION
Base Document Type	Order
Base Process Type	Order Negotiation

**Table A–86 Publish Negotiation Results Attributes**

Attribute	Value
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–87 Publish Negotiation Results Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:

### Pending Job Count

**Table A–88 Publish Negotiation Results Statistics**

Statistic Name	Description
NumNegotiationsProcessed	Number of negotiations processed.
NumNegotiationsPublished	Number of negotiations published.

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.



## Events Raised

The following events are raised by this time-triggered transaction:

**Table A–89 Events Raised by Publish Negotiation Results Transaction**

Base Transaction	Raised when...	Key Data	Data Published	Template Support?
PUBLISH_ORD_NEGOTIATION/ON_SUCCESS	On success	Negotiation_dbd.txt	YCP_getNegotiationDetails_output.xml	Yes *
RECEIVE_ORD_NEGOTIATION/ON_SUCCESS	On success, when DocumentType is 0001, EntityType is ORDER.	Number of concurrent time-triggered transactions running.	receiveOrderNegotiation_dbd.txt	No
<p><b>* Note:</b> Template used for this event is the same template used by the <code>getNegotiationDetails()</code> API to form the output XML.</p>				

## A.2.26 Release

This transaction releases orders to specific ship nodes, making sure that the scheduled ship nodes have enough inventory to process the order.

This transaction should be invoked after the scheduling process.

For more details, see the `releaseOrder()` API in the *Yantra 7x Javadocs*.

---

**Important:** Yantra recommends that if you run the combined 'Schedule and Release' agent, you do not also run the individual Schedule or the individual Release agents.

---

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–90 Release Attributes**

Attribute	Value
Base Transaction ID	RELEASE
Base Document Type	Order
Base Process Type	Order Fulfillment
APIs Called	releaseOrder( )

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–91 Release Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
IgnoreReleaseDate	Optional. Determines whether the schedule process should ignore line release date criteria. Valid values are: <ul style="list-style-type: none"> <li>Y - releases line quantities regardless of release date criteria</li> <li>N - releases lines quantities only after release date criteria have been met. Default.</li> </ul>
CheckInventory	Optional. Determine whether inventory should be checked. Valid values are: <ul style="list-style-type: none"> <li>Y - inventory needs to be checked. Default.</li> <li>N - inventory does not need to be checked</li> </ul>
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–92 Release Criteria Statistics**

Statistic Name	Description
NumFutureDateFailures	Number of orders did not attempt to release because of future date failures.
NumOrdersAttempted	Number of orders attempted to release.
NumOrdersCannotBeProcessedFailures	Number of orders did not attempt to release because of cannot be processed failures.
NumOrdersProcessed	Number of orders processed.
NumOrdersReleased	Number of orders released.
NumOrdersBackordered	Number of orders backordered.
NumOrderLinesReleased	Number of order lines released.
NumOrderLinesBackordered	Number of order lines backordered.
NumReleasesCreated	Number of order releases created.
NumOrdersCannotBeProcessedFailures	Number of orders that were not released due to process failure.

**Note:** If the release process results in splitting of an order line, NumOrderLinesReleased, NumOrderLinesBackordered, and NumOfReleasesCreated may result in more than one count.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table, if tasks on hold are not ready to be processed.

### Events Raised

This transaction raises events specified by the `releaseOrder()` API in the *Yantra 7x Javadocs*.

## A.2.27 Route Shipment

This time-triggered transaction is used to route shipments and belongs to the Outbound Shipment pipeline. It assigns the Carrier and Carrier Service codes for the shipment based on the Routing Guide configured.

The Route Shipment transaction either includes shipments in an existing load or creates a new load and includes the shipments in it.

Shipments can be consolidated to a load, only if the following conditions are met:

- Expected Ship Date - The expected ship date of the shipments must be less than or equal to the must ship before date of the load.
- Expected Load Departure Date - The expected load departure date must be less than or equal to the must ship before date of the shipments in the load.

The must ship before date is a date computed for the load, based on all shipments present in the load. For example, if a load has three shipments with their must ship before dates as 12.22.2005, 12.12.2005, and 12.19.2005 respectively, then the must ship before date of the load is computed as 12.12.2005, as it is the earliest of the three dates.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–93 Route Shipment**

Attribute	Value
Base Transaction ID	ROUTE_SHIPMENT.0001
Base Document Type	Order
Base Process Type	ORDER_DELIVERY
Abstract Transaction	No
APIs Called	None
User Exits Called	com.yantra.ydm.japi.ue.YDMOVERRIDEDETERMINEROUTINGUE com.yantra.ydm.japi.ue.YDMBEFOREDETERMINEROUTINGUE

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–94 Route Shipment Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Route Shipment transaction needs to be run. If not passed, then all enterprises are monitored.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–95 Route Shipment Statistics**

Statistic Name	Description
NumRouted	Number of shipments routed.

## Pending Job Count

For this transaction the pending job count is the number of records representing the unheld orders that are available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

## Events Raised

The following events are raised by this time-triggered transaction:

**Table A–96 Events Raised by the Route Shipment Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
ON_SUCCESS	shipment_dbd.txt	YDM_ROUTE_SHIPMENT.ON_SUCCESS.xml	Yes
ON_FAILURE	shipment_dbd.txt	YDM_ROUTE_SHIPMENT.ON_FAILURE.xml	Yes

However, note that the template name would read <TransactionId>.ON\_SUCCESS.xml.

## A.2.28 Schedule

This transaction schedules orders to specific ship nodes making sure that the scheduled ship nodes have enough inventory to process the order.

Run this transaction after order creation.

**Important:** Yantra recommends that if you run the combined 'Schedule and Release' agent, you do not also run the individual Schedule or the individual Release agents.

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–97 Schedule Attributes**

Attribute	Value
Base Transaction ID	SCHEDULE
Base Document Type	Order
Base Process Type	Order Fulfillment
APIs Called	scheduleOrder( )

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–98** *Schedule Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
OptimizationType	Optional. Determines the optimization rules to apply to the scheduling process. Valid values are: <ul style="list-style-type: none"> <li>• 01 - Optimize on date (Default)</li> <li>• 02 - Optimize on ship node priority</li> <li>• 03 - Optimize on number of shipments</li> </ul>
OrderFilter	Optional. Determines the types of orders to filter. Possible values are: <ul style="list-style-type: none"> <li>• A - All orders (Default)</li> <li>• B - Backorders only</li> <li>• N - New orders only</li> </ul>
ScheduleAndRelease	Optional. Notify the schedule process to release all releasable line quantities. Valid values are: <ul style="list-style-type: none"> <li>• Y - releases successfully scheduled line quantities.</li> <li>• N - only schedules line quantities. Default.</li> </ul> <b>Note:</b> enabling this parameter will not validate hold types configured for the release transaction.

*Table A–98 Schedule Criteria Parameters*

Parameter	Description
IgnoreReleaseDate	Optional. Determines whether the schedule process should ignore line release date criteria. Valid values are: <ul style="list-style-type: none"><li>• Y - releases line quantities regardless of release date criteria</li><li>• N - releases lines quantities only after release date criteria have been met. Default.</li></ul>
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

**Statistics Tracked**

The following statistics are tracked for this transaction:



*Table A–99 Schedule Statistics*

Statistic Name	Description
NumFutureDateFailures	<p>Number of orders that Yantra 7x did not attempt to schedule because of future date failures.</p> <p>Failures can be caused by any of the following:</p> <ul style="list-style-type: none"> <li>• If the OrderFilter is “B” (Backorders Only) and there are no backordered or unscheduled lines.</li> <li>• If the OrderFilter is “N” (New orders Only) and there are some backordered or unscheduled lines.</li> <li>• If order has order lines within only backordered or unscheduled status and the status modify timestamp is after the current time - the back order wait period specified in the scheduling rule.</li> </ul>
NumOrdersAttempted	<p>Number of orders attempted to schedule. This statistic does not include the values for NumFutureDateFailures and NumOrdersCannotBeProcessedFailures statistics.</p>
NumOrderLinesReleased	<p>Number of order lines that have been released.</p>

Table A-99 Schedule Statistics

Statistic Name	Description
NumOrdersCannotBeProcessedFailures	<p>Number of orders that Yantra 7x did not attempt to schedule because of cannot be processed failures.</p> <p>Failures can be caused by any of the following:</p> <ul style="list-style-type: none"> <li>• The result of the YFSCheckOrderBeforeProcessingUE user exit returns as false.</li> <li>• The Order has the HoldFlag attribute set to 'Y'.</li> <li>• The Order has the SaleVoided attribute set to 'Y'.</li> <li>• The Order does not have PaymentStatus as AUTHORIZED, INVOICED, PAID, nor NOT_APPLICABLE.</li> </ul>
NumOrdersCreated	Number of orders created. This also includes the number of procurement orders created.
NumOrderLinesCreated	Number of order lines created.
NumOrdersProcessed	Number of orders processed.
NumOrdersScheduled	Number of orders that have at least one line that was scheduled. <b>Note:</b> scheduled includes all status changes except BACKORDER.
NumOrdersProcOrdersCreated	Number of procurement orders created.
NumWorkOrdersCreated	Number of work orders created.
NumOrdersBackordered	Number of orders backordered.
NumOrderLinesScheduled	Number of order lines scheduled.
NumOrderLinesBackordered	Number of order lines backordered.
NumReleasesCreated	Number of order releases created.

### Pending Job Count

For this transaction the pending job count is the number of records representing the unheld orders that are available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table, if tasks on hold are not ready to be processed.

### Events Raised

This transaction raises events as specified in the `scheduleOrder()` API in *Yantra 7x Javadocs*.

## A.2.29 Send Invoice

This transaction publishes invoice data that can be directed to an external accounts receivable systems.

In environments that require an interface with accounts receivable systems, this job needs to be scheduled. This job raises an event for an invoice based on the following configuration at the following times in the order lifecycle:

- Publish invoice at shipment creation - This implies that your accounts payable system will take care of payment collection. Invoices can be published as soon as they are created.
- Publish invoice after payment collection - This implies that the Yantra 7x Application Consoles takes care of the payment collection and an invoice notification can be received only after the total order amount is collected.

---



---

**Note:** Many of this transaction's elements and attributes are template driven. Refer to the XML for element level details.

---



---

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–100 Send Invoice Attributes**

Attribute	Value
Base Transaction ID	SEND_INVOICE
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	getOrderInvoiceDetails()

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–101 Send Invoice Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–102 Send Invoice Statistics**

Statistic Name	Description
NumInvoicesSent	Number of invoices sent.

### Pending Job Count

For this transaction the pending job count is the number of order invoices in created ("00") status.

### Events Raised

The following events are raised by this time-triggered transaction:

**Table A–103 Events Raised by the Send Invoice Transaction**

Transaction/Event	Key Data	Data Published	Template Support?
PUBLISH_INVOICE_DETAIL	modifyOrder_dbd.txt and sendInvoice_dbd.txt	YFS_getOrderInvoiceDetails_output.xml	Yes

Additional events may be raised by the `getOrderInvoiceDetails()` API. For detailed information about the events, see the *Yantra 7x Javadocs* for this API.

### A.2.30 Send Order

This transaction tries to raise the ON\_SUCCESS event for an order whose OrderHeaderKey is stored in the task queue object. The event is raised only if all of the order lines of the order reach particular status(es) completely. That is, the entire ORDERED\_QTY of each line must be in the particular status(es). In addition to raising the event, the line statuses are also changed to the drop statuses, corresponding to the pickup statuses. The SendOrder transaction, derived from the abstract transaction SEND\_ORDER, should have the event, pickup, and drop statuses configured. For more information, see the `sendOrder()` API in *Yantra 7x Javadocs*.

If an order needs to be communicated to a third party, use this transaction.

---



---

**Note:** The TransactionKey posted in the task object must be an instance of the Abstract Transaction SEND\_ORDER for the ProcessType associated with the Order. Otherwise, an exception is thrown.

---



---

#### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–104** *Send Order Attributes*

Attribute	Value
Base Transaction ID	SEND_ORDER
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	Yes
APIs Called	<code>sendOrder()</code>

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–105** *Send Order Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

None.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the `AVAILABLE_DATE` value less than or equal to ( $\leq$ ) the current date value in the `YFS_Task_Q` table.

### Events Raised

This transaction raises events specified by the `sendOrder()` API in the *Yantra 7x Javadocs*.

### A.2.31 Send Release

The Send Release Agent dispatches releases to ship nodes.

#### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–106 Send Release Attributes**

Attribute	Value
Transaction Name	Send Release
Transaction ID	SHIP_ADVICE
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	com.yantra.yfs.agent.YFSWMSShipAdviceAgent

#### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–107 Send Release Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

#### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–108 Send Release Statistics**

Statistic Name	Description
NumReleasesProcessed	Number of order releases processed.
NumReleasesSent	Number of order releases sent.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

### Events Raised

The following events are raised by this time-triggered transaction:

**Table A–109 Events Raised by the Send Release Transaction**

Transaction/Event	Data Published
PUBLISH_SHIP_ADVICE	YFS_publishShipAdvice.xml

## A.2.32 Start Order Negotiation

This transaction creates the negotiations for orders that are configured to go through the negotiation process.

Use this transaction in environments where an Order needs to go through a Negotiation phase before it is released.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–110 Start Order Negotiation Attributes**

Attribute	Value
Base Transaction ID	START_ORD_NEGOTIATION
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No



**Table A–110 Start Order Negotiation Attributes**

Attribute	Value
APIs Called	createNegotiation()
User Exits Called	YCPBeforeCreateNegotiationUE, YCPGetNegotiationNoUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–111 Start Order Negotiation Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–112 Start Order Negotiation Statistics**

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumNegotiationsCreated	Number of negotiations created.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

### Events Raised

This transaction raises events specified by the `createNegotiation()` API in the *Yantra 7x Javadocs*.

## A.3 Time-Triggered Purge Transactions

There are several transactions that you can use to purge your database tables at specific time intervals.

Purge transactions determine when a table should be purged by determining the current date and subtracting the retention days specified by the purge. If the timestamp on the table is less than or equal to (current day - retention days) the table is purged.

---

---

**Note:** In some cases, a purge may look at another field other than the table's timestamp. These are pointed out in the documentation.

---

---

---

---

**Note:** When an entity is being purged, the related or dependent information that is present in other tables should be taken into consideration for purging along with it. For example, if a sales order with live shipments is being purged, any cross reference to that order will not be accurate in the Order Shipment Console.

---

---

---

**Note:** Some of the statistics collected and tracked in Release 7.5 SP1 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Yantra.

---



---

**Note:** All Time-Triggered Purge Transactions have a `CollectPendingJobs` criteria parameter. If this parameter is set to `N`, the agent will not collect information on the pending jobs for that time-triggered transaction. This pending job information is used for monitoring the agent in the *Yantra 7x System Management Guide*.

By default, `CollectPendingJobs` is set to `Y`. It can be helpful to set it to `N` if one particular time-triggered transaction is performing a significant amount of `getPendingJobs` queries, and the overhead cost is too high.

---

### A.3.1 Purge Strategy

The following recommendations should be taken into consideration when planning a purge strategy for each purge transaction:

- Test purges by setting Live to 'N'.
- Turn on logging to test what is purged.
- Set up purge traces in the System Management Console and analyze the information.

### A.3.2 Configuring Purge Transaction Log Files

You can configure purges to write log files to a directory you specify. Each time you run a particular purge, new data is appended to this file. If no file exists, one is created.

To specify purge log file directory:

1. Edit the `<YFS_HOME>/resources/yfs.properties` file.
2. Set `yfs.purge.path=<full absolute path of log directory>`.

For example, on UNIX you might specify the log files to be written to the `/app/yfs/logs/purges` directory.

### A.3.3 Available Purges

This section contains details of all purge transactions in alphabetical order. The time-triggered purge transactions are:

- [Alert Purge](#)
- [Capacity Purge](#)
- [Delivery Plan Purge](#)
- [Export Table Purge](#)
- [Import Table Purge](#)
- [Inventory Purge](#)
- [Inventory Audit Purge](#)
- [Load Purge](#)
- [Manifest Purge](#)
- [Negotiation History Purge](#)
- [Negotiation Purge](#)
- [Order History Purge](#)
- [Order Purge](#)
- [Order Status Audit Purge](#)
- [Picklist Purge](#)
- [Price List Purge](#)
- [Receipt History Purge](#)
- [Receipt Purge](#)
- [Reprocess Error Purge](#)
- [Reservation Purge](#)
- [Shipment History Purge](#)
- [Shipment Purge](#)

- [Shipment Statistics Purge](#)
- [Statistics Purge](#)
- [Work Order History Purge](#)
- [Work Order Purge](#)

### A.3.3.1 Alert Purge

This purge removes alert data from the system. This reduces the load from frequently accessed tables. The alert should be marked as CLOSED.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

All inactive alerts with the resolution date smaller than or equal to the current date minus the purge criteria's retention days can be configured to be picked up by the Alert Purge.

This purge agent also closes any open alerts where the number of expiration days is greater than zero, and the modified timestamp is less than the current date minus the number of expiration days.

The alert purge agent purges only the alerts that are marked as CLOSED.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A-113 Alert Console Purge Attributes**

Attribute	Value
Base Transaction ID	INBOXPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–114 Alert Console Purge Criteria Parameters**

Criteria Parameters	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
CollectPendingJobs	If this parameter is set to N, the agent will not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in Yantra 7x System Management.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. The organization for which the Alert Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–115 Alert Console Purge Statistics**

Statistic Name	Description
NumInboxPurged	Number of inbox records purged.

**Pending Job Count**

For this transaction the pending job count is the number of records that can be purged from the YFS\_INBOX table.

**Events Raised**

None.

**Tables Purged**

YFS\_INBOX

YFS\_INBOX\_AUDIT

YFS\_INBOX\_REFERENCES

**A.3.3.2 Capacity Purge**

This purge removes capacity data from the system. This reduces load from the frequently accessed tables.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

The following can be configured to be picked up by the Capacity Purge:

- All resource pool standard capacity periods with effective to date less than or equal to the current date minus the purge criteria's retention days.
- All resource pool overridden capacity with the capacity date less than or equal to current date minus the purge criteria's retention days.
- All resource pool capacity consumption with consumption date less than or equal to the current date minus the purge criteria's retention days.
- All capacity reservations where appointment date is less than system date minus the purge criteria's retention days (or ManualReservationPurgeLeadDays for manual reservations) and reservation Id is blank.
- All capacity reservations where expiration date has passed and reservation Id is not blank.

**Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–116 Capacity Purge Attributes**

Attribute	Value
Base Transaction ID	CAPACITYPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–117 Capacity Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:



**Table A-118 Capacity Purge Statistics**

Statistic Name	Description
NumStdCapacityPeriodsPurged	Number of standard capacity periods purged.
NumCapacityOverridesPurged	Number of capacity overrides purged.
NumCapacityConsumptionsPurged	Number of capacity consumptions purged.

### Pending Job Count

For this transaction the pending job count is the total number of records that can be purged from the YFS\_RES\_POOL\_STD\_CAPCTY\_PERD, YFS\_RES\_POOL\_CAPCTY\_OVERRIDE, YFS\_RES\_POOL\_CONSMPTN\_DTLS and YFS\_RES\_POOL\_CAPCTY\_CONSMPTN tables.

### Events Raised

None.

### Tables Purged

The YFS\_RES\_POOL\_STD\_CAPCTY\_PERD table is purged when  $\text{EFFECTIVE\_TO\_DATE} \leq (\text{CurrentDate} - \text{LeadDays})$

The YFS\_RES\_POOL\_CAPCTY\_OVERRIDE table is purged when  $\text{CAPACITY\_DATE} \leq (\text{CurrentDate} - \text{LeadDays})$

The YFS\_RES\_POOL\_CAPCTY\_CONSMPTN table is purged when  $\text{CONSUMPTION\_DATE} \leq (\text{CurrentDate} - \text{LeadDays})$ , or if a manual reservation is taken, when  $\text{CONSUMPTION\_DATE} \leq (\text{CurrentDate} - \text{Manual Reservation Lead Days})$ . When this table is purged, YFS\_RES\_POOL\_CONSMPTN\_DTLS will be purged also.

The YFS\_RES\_POOL\_CONSMPTN\_DTLS table is purged when  $\text{RESERVATION\_EXPIRATION\_DATE} \leq (\text{CurrentDate} - \text{LeadDays})$

### A.3.3.3 Delivery Plan Purge

This purge deletes delivery plans after they have completed their typical lifecycles. It purges all the delivery plans that have been marked as 'Closed' for a period greater than the retention days specified in the

criteria parameters and those that do not have any shipments or loads. The order should have been moved to history before the lead time setup.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

**Note:** All the loads and shipments that are associated with the delivery plans should have been purged before running this purge agent.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–119** *Delivery Plan Purge Attributes*

Attribute	Value
Base Transaction ID	DELIVERYPLANPRG
Base Document Type	Load
Base Process Type	Load Execution
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–120** *Delivery Plan Purge Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

**Table A–120 Delivery Plan Purge Criteria Parameters**

Parameter	Description
EnterpriseCode	Optional. Enterprise for which the Delivery Plan Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Defaults to N. <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–121 Delivery Plan Purge Statistics**

Statistic Name	Description
NumDeliveryPlansPurged	Number of delivery plans purged.

### Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Delivery\_Plan table.

### Events Raised

None.

### Tables Purged

YFS\_DELIVERY\_PLAN

A.3.3.4 Export Table Purge

This purge removes export table data from the system. This reduces load from the frequently accessed tables. It purges records in YFS\_EXPORT tables that meet the following criteria:

- YFS\_EXPORT records should be marked as processed (Status = 10).
- The last modified time should be before the lead time setup.

**Note:** This purge only reads the rules defined by the hub. Enterprise overridden rules are not considered.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–122 Export Table Purge Attributes

Attribute	Value
Base Transaction ID	EXPORTTBLPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–123 Export Table Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–124 Export Table Purge Statistics**

Statistic Name	Description
NumExportsPurged	Number of exports purged.

## Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Export table.

## Events Raised

None.

Tables Purged

YFS\_EXPORT

A.3.3.5 Import Table Purge

This purge removes import table data from the system. This reduces load from the frequently accessed tables. It purges records in YFS\_IMPORT tables that meet the following criteria:

- YFS\_IMPORT records should be marked as processed (Status = "10").
- The "last modified time" should be before the lead time setup.

**Note:** This purge only reads the rules defined by the hub. Enterprise overridden rules are not considered.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–125 Import Table Purge Attributes

Attribute	Value
Base Transaction ID	IMPORTTBLPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–126 Import Table Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–127 Import Table Purge Statistics**

Statistic Name	Description
NumImportsPurged	Number of import tables purged.

### Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Import table.

### Events Raised

None.

### Tables Purged

YFS\_IMPORT

A.3.3.6 Inventory Audit Purge

This purge removes inventory audit data from the system. This reduces load from the frequently accessed tables.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

All inventory audits of the provided organization with modify timestamp less than the current date minus the purge criteria’s retention days can be configured to be picked up by the Inventory Audit Purge.

**Note:** Number of Threads for this purge’s agent criteria details must be set to 1. For more information on agent criteria, refer to the *Yantra 7x Platform Configuration Guide*.

**Important:** The Inventory Audit purge will not purge any records under 60 days old, even if configured to do so.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–128 Inventory Audit Purge Attributes

Attribute	Value
Base Transaction ID	INVENTORYAUDITPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE



## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–129 Inventory Audit Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. The inventory organization for which the Inventory Audit Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Table Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–130 Inventory Audit Statistics**

Statistic Name	Description
NumInventoryAuditsPurged	Number of inventory audits purged.

## Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Inventory\_Audit table.

Events Raised

None.

Table Purged

YFS\_INVENTORY\_AUDIT

A.3.3.7 Inventory Purge

This purge removes inventory data from the system. This reduces load from the frequently accessed tables. This purge does not take retention days into account when purging.

You can use purge codes pseudo-logic to benefit in analyzing purges. An inventory data is picked up for purge if it meets the following criteria:

- Supply record has the same availability type as the node. For example, TRACK or INFINITE.
- Supply record has 0 quantity.
- Supply record do not contain the supply type "INFO".

For YFS\_INVENTORY\_TAG, it is purged if the INVENTORY\_TAG\_KEY is not used by any of the existing supply and demand, with the following two exceptions:

- Ship Node Inventory Type is Infinite, Inventory Supply Type has Track, and Quantity = 0, then not purged.
- Ship Node Inventory Type is Track, Inventory Supply Type has Infinite, and Quantity = 0, then not purged.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–131 Inventory Purge Attributes

Attribute	Value
Base Transaction ID	INVENTORYPRG
Base Document Type	General

**Table A–131 Inventory Purge Attributes**

Attribute	Value
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–132 Inventory Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–133 Inventory Purge Statistics**

Statistic Name	Description
NumInventoryDemandsPurged	Number of inventory demands purged.
NumInventoryReservationsPurged	Number of inventory reservations purged.
NumInventoryTagsPurged	Number of inventory tags purged.

**Pending Job Count**

For this transaction the pending job count is the total number of records that can be purged from the YFS\_Inventory\_Supply, YFS\_Inventory\_Demand, YFS\_Inventory\_Tag, and YFS\_Inventory\_Reservation tables.

**Events Raised**

None.

**Tables Purged**

YFS\_INVENTORY\_DEMAND

YFS\_INVENTORY\_TAG

YFS\_INVENTORY\_RESERVATION

YFS\_INVENTORY\_SUPPLY

**A.3.3.8 Inventory Supply Temp Purge**

The Inventory Supply Temp purge agent cleans up the contents in the temporary inventory tables generated by the process of synchronizing Yantra 7x's inventory picture with the actual inventory picture at the nodes.

The node inventory picture is stored during the loading process into the YFS\_INVENTORY\_SUPPLY\_TEMP table. Once the synchronization phase is complete and the YFS\_INVENTORY\_SUPPLY table has been updated, the YFS\_INVENTORY\_SUPPLY\_TEMP table needs to be purged, which is done through this agent.

For more information on configuring the synchronization with node inventory, refer to the *Yantra 7x Inventory Synchronization Configuration Guide*.

The Inventory Supply Temp purge agent will be used to purge all records in the YFS\_INVENTORY\_SUPPLY\_TEMP table whose modify timestamp is less than current time minus purge criteria's lead days for a group of YantraMessageGroupID.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–134 Inventory Supply Temp Purge Attributes**

Attribute	Value
Base Transaction ID	SUPPLYTEMPPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–135 Inventory Supply Temp Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

**Table A–135 Inventory Supply Temp Purge Criteria Parameters**

Parameter	Description
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
EnterpriseCode	Optional. The inventory organization for which the Inventory Supply Temp Purge needs to be run. If not passed, then all enterprises are monitored.organization.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–136 Inventory Supply Temp Purge Statistics**

Statistic Name	Description
NumInventorySupplyTempsPurged	Number of entries in the YFS_INVENTORY_SUPPLY_TEMP table purged.

### Pending Job Count

Number of unique YantraMessageGroupIDs from YFS\_INVENTORY\_SUPPLY\_TEMP table whose maximum modify timestamp is less than current timestamp minus purge criteria's lead day.

### Events Raised

None.

### Tables Purged

YFS\_INVENTORY\_SUPPLY\_TEMP

### A.3.3.9 Load Purge

This purge removes load data from the system. It picks up all loads that have been marked as 'Closed' and purges them. Empty Loads (for example, loads with no shipments) are not considered for purge. As a part of this purge, the associated child tables are also purged.

This is not a pipeline transaction. It also does not work from the task queue.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

#### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–137 Load Purge Attributes**

Attribute	Value
Base Transaction ID	LOADPRG
Base Document Type	Load
Base Process Type	Load Execution
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

#### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–138 Load Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

**Table A–138 Load Purge Criteria Parameters**

Parameter	Description
EnterpriseCode	Optional. Enterprise for which the Load Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–139 Load Purge Statistics**

Statistic Name	Description
NumLoadShipmentsPurged	Number of load shipments purged.
NumLoadsPurged	Number of loads purged.

### Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Load table.

### Events Raised

None.

### Tables Purged

YFS\_ADDITIONAL\_DATE

YFS\_LOAD



YFS\_LOAD\_STOP

YFS\_LOAD\_SHIPMENT

YFS\_LOAD\_SHIPMENT\_CHARGES (charges that pertain to this load)

YFS\_LOAD\_STATUS\_AUDIT

YFS\_LOADED\_CONTAINER

YFS\_SHIPMENT\_CONTAINER

YFS\_CONTAINER\_ACTIVITY

### A.3.3.10 Manifest Purge

This purge picks up all the manifests that have been marked as 'Closed' and purges them.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

All manifests which do not associate to any shipments and with modify timestamp less than the current date minus the purge criteria's retention days can be configured to be picked up by the Manifest Purge.

**Note:** All the shipments associated with the manifests should have been purged before running this purge agent.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–140 Manifest Purge Attributes**

Attribute	Value
Base Transaction ID	MANIFESTPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–141 Manifest Purge Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that will only perform their tasks on the nodes with a matching node transactional velocity value.  Valid values are: LOW, HIGH, and any additional values defined by the Hub from Platform > System Administration > Agent Criteria Groups.
ShipNode	Optional. Ship node for which the Manifest Purge needs to be run. If not passed, then all ship nodes are monitored.

## Statistics Tracked

The following statistics are tracked for this transaction:

*Table A–142 Manifest Purge Statistics*

Statistic Name	Description
NumManifestsPurged	Number of manifests purged.

**Pending Job Count**

For this transaction the pending job count is the number of records that can be purged from the YFS\_Manifest table.

**Events Raised**

None.

**Tables Purged**

YFS\_MANIFEST

**Note:** To purge Manifests, the Shipments related to the manifests should be purged by Shipment Purge, and the Manifests should be in 'Closed' status. For more information, see [Section A.3.3.23, "Shipment Purge"](#) on page 228.

**A.3.3.11 Negotiation History Purge**

This purge removes negotiation history data from the system. This reduces load from the frequently accessed tables. It purges data from the order negotiation history tables.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

**Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–143** *Negotiation History Purge Attributes*

Attribute	Value
Base Transaction ID	NEGOTIATIONHISTPRG
Base Document Type	Order
Base Process Type	Order Negotiation
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–144** *Negotiation History Purge Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Negotiation History Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–145** *Negotiation History Purge Statistics*

Statistic Name	Description
NumNegotiationHistoriesPurged	Number of negotiation histories purged.

## Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Negotiation\_Hdr\_H table.

## Events Raised

None.

## Tables Purged

YFS\_NEGOTIATION\_HDR\_H

YFS\_NEGOTIATION\_LINE\_H

YFS\_RESPONSE\_H

YFS\_RESPONSE\_HDR\_H

YFS\_RESPONSE\_LINE\_H

YFS\_RESPONSE\_LINE\_DTL\_H

### A.3.3.12 Negotiation Purge

This purge archives data into history tables after it completes its typical lifecycle. This reduces load from the frequently accessed tables. It works from the task queue (YFS\_TASK\_Q) table, picking up negotiations in which the last modification time occurs before the lead time set up and which are in statuses that can be picked.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–146** *Negotiation Purge Attributes*

Attribute	Value
Base Transaction ID	ORD_NEGOTIATION_PURGE
Base Document Type	Order
Base Process Type	Order Negotiation
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–147** *Negotiation Purge Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Negotiation Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>

**Table A–147** *Negotiation Purge Criteria Parameters*

Parameter	Description
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–148** *Negotiation Purge Statistics*

Statistic Name	Description
NumOrderNegotiationsPurged	Number of order negotiations purged.

### Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

### Events Raised

None

### Tables Purged

YFS\_NEGOTIATION\_HDR

YFS\_NEGOTIATION\_LINE

YFS\_RESPONSE

YFS\_RESPONSE\_HDR

YFS\_RESPONSE\_LINE

YFS\_RESPONSE\_LINE\_DTL

A.3.3.13 Order History Purge

This purge deletes data from history tables after it completes its typical lifecycle. This reduces load from the frequently accessed tables. It deletes data from the history tables.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

**Note:** Order should have been purged and moved into the history tables before you can run this transaction. See [Section A.3.3.14, "Order Purge"](#) on page 206.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–149 Order History Purge Attributes

Attribute	Value
Base Transaction ID	ORDERHISTPRG
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–150 Order History Purge Criteria Parameters

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.



**Table A–150 Order History Purge Criteria Parameters**

Parameter	Description
EnterpriseCode	Optional. Enterprise for which the Order History Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Removes qualifying records from the history tables listed under <a href="#">Tables Purged</a>.</li> <li>N- Test mode. Determines the rows that will be removed without actually removing them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–151 Order History Purge Statistics**

Statistic Name	Description
NumOrderHistoriesPurged	Number of order histories purged.

### Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Order\_Header\_H table.

### Events Raised

None.

### Tables Purged

YFS\_CHARGE\_TRANSACTION\_H

YFS\_CREDIT\_CARD\_TRANSACTION\_H

YFS\_ENTITY\_ADDRESS\_H

YFS\_HEADER\_CHARGES\_H

YFS\_INSTRUCTION\_DETAIL\_H  
YFS\_INVOICE\_COLLECTION\_H  
YFS\_LINE\_CHARGES\_H  
YFS\_NOTES\_H  
YFS\_ORDER\_AUDIT\_DETAIL\_H  
YFS\_ORDER\_AUDIT\_H  
YFS\_ORDER\_AUDIT\_LEVEL\_H  
YFS\_ORDER\_DATE\_H  
YFS\_ORDER\_HEADER\_H  
YFS\_ORDER\_HOLD\_TYPE\_H  
YFS\_ORDER\_HOLD\_TYPE\_LOG\_H  
YFS\_ORDER\_INVOICE\_DETAIL\_H  
YFS\_ORDER\_INVOICE\_H  
YFS\_ORDER\_KIT\_LINE\_H  
YFS\_ORDER\_KIT\_LINE\_SCHEDULE\_H  
YFS\_ORDER\_LINE\_H  
YFS\_ORDER\_LINE\_OPTION\_H  
YFS\_ORDER\_LINE\_REQ\_TAG\_H  
YFS\_ORDER\_LINE\_SCHEDULE\_H  
YFS\_ORDER\_PROD\_SER\_ASSOC\_H  
YFS\_ORDER\_RELEASE\_H  
YFS\_ORDER\_RELEASE\_STATUS\_H  
YFS\_ORDER\_SER\_PROD\_ITEM\_H  
YFS\_PAYMENT\_H  
YFS\_REFERENCE\_TABLE\_H  
YFS\_TAX\_BREAKUP\_H

### A.3.3.14 Order Purge

This purge archives data into history tables after it completes its typical lifecycle. See [Section A.3.3.13, "Order History Purge"](#) on page 203. This reduces load from the frequently accessed tables. It works on a task queue. It picks up the orders from YFS\_TASK\_Q table that are available for the transaction PURGE.

**Note:** This transaction depends on all lines of an order being in a status pickable by the Purge transaction.

The following statuses are available for configuration to be picked up by Order Purge:

- Draft Created (1000) and all extended Draft Created Statuses.
- Created (1100) and all extended Created statuses. These statuses are available only for document types Sales Order, Purchase Order and Transfer Order.
- Shipped (3700) and all extended Shipped statuses.
- Cancelled (9000) and all extended Cancelled statuses.
- Shorted (9020) and all extended Shorted statuses.

You can use purge codes pseudo-logic to benefit in analyzing purges. An order is picked up for purge if it meets the following criteria:

1. All open child orders (derived, chained, return, exchange, procurement, or refund fulfillment) for the order must already be purged.
2. No pending transfer-out charges to another order exceeding the transfer-ins.
3. No pending adjustment invoices.

An order is purged immediately if it meets the above three criteria and is completely cancelled.

If an order does not meet any of the above criteria continue checking for the criteria given below:

- No order release status record that does not meet the lead days.
- It should be in the correct status for purge. For example,

- All service requests for the order should have Shipped or extended Shipped status.
  - The payment status for the order should be Paid, Cancelled, or Not Applicable.
  - It must not have any unpurged negotiations.
- For all order lines other than service request lines:
  - If the Seller inventory update is required, the Status Inventory Type has the "Update Seller Supply" option turned on, and the Seller Supply Type is "Onhand", or blank. (The Seller Supply Type can also be a custom seller supply type with the "Onhand Supply" checkbox enabled.)
  - If the Seller Demand Type is blank.
  - If the Buyer inventory update is required and the Buyer Supply Type is "Onhand", or blank.
- The order's last modification should fall before the lead time setup.
- Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.
- The order must not have a undelivered service line.
- In the case of an exchange order for processing a return order, the exchange order should be purged from history before the return order can be purged.

**Note:** With no change to status inventory type, a sales order in Shipped (3700) status or its extended status is purged if the Buyer is not passed.

An order in Shipped status or extended Shipped status in the default pipeline is not purged if the Buyer passed on the sales order is tracking inventory. This prevents the purging of the order relating to the pending supply for the Buyer tracking inventory.

To purge such orders, the status inventory type for the Shipped or extended Shipped status should be configured such that the Buyer Supply Type is ONHAND for the status inventory type.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–152 Order Purge Attributes**

Attribute	Value
Base Transaction ID	PURGE
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–153 Order Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Next Task Queue Interval	Optional. Specifies in hours how long a failed task should be suspended before it is considered for reprocessing. Defaults to 5 hours.
EnterpriseCode	Optional. Enterprise for which the Order Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–154 Order Purge Statistics**

Statistic Name	Description
NumOrdersProcessed	Number of order processed.
NumOrdersPurged	Number of orders purged.

**Pending Job Count**

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

**Events Raised**

None.

**Tables Purged**

YFS\_CHARGE\_TRANSACTION  
 YFS\_CREDIT\_CARD\_TRANSACTION  
 YFS\_ENTITY\_ADDRESS  
 YFS\_HEADER\_CHARGES  
 YFS\_INSTRUCTION\_DETAIL  
 YFS\_INVOICE\_COLLECTION  
 YFS\_LINE\_CHARGES  
 YFS\_MONITOR\_ALERT  
 YFS\_NOTES  
 YFS\_ORDER\_AUDIT  
 YFS\_ORDER\_AUDIT\_DETAIL  
 YFS\_ORDER\_AUDIT\_LEVEL  
 YFS\_ORDER\_HEADER  
 YFS\_ORDER\_HOLD\_TYPE  
 YFS\_ORDER\_HOLD\_TYPE\_LOG  
 YFS\_ORDER\_INVOICE  
 YFS\_ORDER\_INVOICE\_DETAIL  
 YFS\_ORDER\_KIT\_LINE  
 YFS\_ORDER\_KIT\_LINE\_SCHEDULE  
 YFS\_ORDER\_LINE

YFS\_ORDER\_LINE\_OPTION  
YFS\_ORDER\_LINE\_REQ\_TAG  
YFS\_ORDER\_LINE\_SCHEDULE  
YFS\_ORDER\_LINE\_SRC\_CNTRL  
YFS\_ORDER\_PROD\_SER\_ASSOC  
YFS\_ORDER\_RELEASE  
YFS\_ORDER\_RELEASE\_STATUS  
YFS\_ORDER\_SER\_PROD\_ITEM  
YFS\_ORDER\_DATE  
YFS\_PAYMENT  
YFS\_REFERENCE\_TABLE  
YFS\_TAX\_BREAKUP  
YFS\_ACTIVITY\_DEMAND

**A.3.3.15 Order Status Audit Purge**

This purge removes order status audit data from the system. This reduces load from the frequently accessed tables. It purges all records before the lead time setup.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

**Note:** This transaction needs to be run after negotiation is completed.

**Attributes**

The following are the attributes for this time-triggered transaction:

*Table A–155 Order Status Audit Purge Attributes*

Attribute	Value
Base Transaction ID	STATUSAUDITPRG
Base Document Type	Order



**Table A–155 Order Status Audit Purge Attributes**

Attribute	Value
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–156 Order Status Audit Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Order Status Audit Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

### Pending Job Count

**Table A–157** *Order Status Audit Purge Statistics*

Statistic Name	Description
NumStatusAuditsPurged	Number of status audits purged.

For this transaction the pending job count is the number of records that can be purged from the YFS\_Status\_Audit table.

### Events Raised

None.

### Tables Purged

YFS\_STATUS\_AUDIT

### A.3.3.16 Picklist Purge

This purge picks up all picklists that have been existing for a period greater than the retention days specified in the criteria parameters and those that do not have any shipments.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

### Attributes

**Note:** All shipments associated with the picklists should have been purged before running this purge agent.

The following are the attributes for this time-triggered transaction:

**Table A–158** *Picklist Purge Attributes*

Attribute	Value
Base Transaction ID	PICKLISTPRG
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	No

**Table A–158 Picklist Purge Attributes**

Attribute	Value
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–159 Picklist Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–160 Picklist Purge Statistics**

Statistic Name	Description
NumPickListsPurged	Number of picklists purged.

**Pending Job Count**

For this transaction the pending job count is the number of records that can be purged from the YFS\_Pick\_List table.

**Events Raised**

None.

**Tables Purged**

YFS\_PICK\_LIST

**A.3.3.17 Price List Purge**

This purge removes price list data from the system. This reduces load from the frequently accessed tables.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

All price list sets with valid date less than the current date minus the purge criteria’s retention days can be configured to be picked up by the Price List Purge.

**Attributes**

The following are the attributes for this time-triggered transaction:

*Table A–161 Price List Purge Attributes*

Attribute	Value
Base Transaction ID	PRICELISTPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–162 Price List Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–163 Price List Purge Statistics**

Statistic Name	Description
NumPriceSetsPurged	Number of price sets purged.

## Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Price\_Set table.

## Events Raised

None.

Tables Purged

YFS\_PRICE\_SET table with VALID\_TILL\_DATE less than or equal to (CurrentDate - LeadDays)

YFS\_PRICE\_PROGRAM\_DEFN

YFS\_ITEM\_PRICE\_SET

YFS\_ITEM\_PRICE\_SET\_DTL

A.3.3.18 Receipt History Purge

This transaction deletes receipts previously archived by the Receipt Purge. See [Section A.3.3.19, "Receipt Purge"](#) on page 219.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

**Note:** To purge a receipt history, ensure that the Receipts are closed and Shipments are purged.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–164    *Receipt History Purge Attributes*

Attribute	Value
Base Transaction ID	RECEIPTHISTPRG
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–165    Receipt History Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Receipt History Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Removes qualifying records from the history tables listed under <a href="#">Tables Purged</a>.</li> <li>N- Test mode. Determines the rows that will be removed without actually removing them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–166    Receipt History Purge Statistics**

Statistic Name	Description
NumReceiptLineHistoriesPurged	Number of receipt line histories purged.
NumReceiptHistoriesPurged	Number of receipt histories purged.

### Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Receipt\_Header\_H table.

### Events Raised

None.

Tables Purged

- YFS\_RECEIPT\_HEADER\_H
- YFS\_RECEIPT\_LINE\_H
- YFS\_RECEIPT\_STATUS\_AUDIT\_H
- YFS\_INSTRUCTION\_DETAIL\_H

A.3.3.19 Receipt Purge

This purge removes receipt data from the system. This reduces load from the frequently accessed tables. This transaction picks up all receipts that are not open and not pending inspection and archives them into their history tables. See [Section A.3.3.18, "Receipt History Purge"](#) on page 217. It also archives and purges the receipt's child tables.

This is a pipeline transaction and works from a task queue.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

**Note:** To purge a receipt, ensure that the Receipts are closed and Shipments are purged.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–167 Receipt Purge Attributes

Attribute	Value
Base Transaction ID	RECEIPTPRG
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE



## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–168** *Receipt Purge Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Receipt Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–169** *Receipt Purge Statistics*

Statistic Name	Description
NumReceiptLinesPurged	Number of Receipt Lines purged.
NumReceiptsPurged	Number of receipts purged.

## Pending Job Count

For this transaction the pending job count is the number of records available to be processed by the transaction with the AVAILABLE\_DATE

value less than or equal to ( $\leq$ ) the current date value in the YFS\_Task\_Q table.

Events Raised

None.

Tables Purged

YFS\_RECEIPT\_HEADER

YFS\_RECEIPT\_LINE

YFS\_RECEIPT\_STATUS\_AUDIT

YFS\_INSTRUCTION\_DETAIL

A.3.3.20 Reprocess Error Purge

This purge deletes reprocess errors from the system. This reduces load from the frequently accessed tables. It purges records in YFS\_REPROCESS\_ERROR tables that meet the following criteria:

- YFS\_REPROCESS\_ERROR records with State = Fixed or Ignored are processed.
- The last modified time should be earlier than the lead time setup.

**Note:** This purge only reads the rules defined by the hub. Enterprise overridden rules are not considered.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–170 Reprocess Error Purge Attributes

Attribute	Value
Base Transaction ID	REPROCESSPRG
Base Document Type	General

**Table A–170 Reprocess Error Purge Attributes**

Attribute	Value
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–171 Reprocess Error Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–172 Reprocess Error Purge Statistics**

Statistic Name	Description
NumReprocessErrsPurged	Number of reprocess errors purged.

**Pending Job Count**

For this transaction the pending job count is the number of records that can be purged from the YFS\_REPROCESS\_ERROR table.

**Events Raised**

None.

**Tables Purged**

YFS\_REPROCESS\_ERROR

**A.3.3.21 Reservation Purge**

This purge deletes expired inventory reservations from the system. This reduces load from the frequently accessed tables as well as free up demands that are consumed by expired reservations. It purges records in YFS\_INVENTORY\_RESERVATION tables that meet the following criteria:

- Records with EXPIRATION\_DATE earlier than current date.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

**Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–173 Reservation Purge Attributes**

Attribute	Value
Base Transaction ID	RESERVATIONPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No

**Table A–173 Reservation Purge Attributes**

Attribute	Value
APIs Called	None
User Exits Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–174 Reservation Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–175 Reservation Purge Statistics**

Statistic Name	Description
NumReservationsPurged	Number of reservations purged.

Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_INVENTORY\_RESERVATION table.

Events Raised

None.

Tables Purged

YFS\_INVENTORY\_RESERVATION

A.3.3.22 Shipment History Purge

This transaction deletes shipments previously archived by the Shipment Purge. See [Section A.3.3.23, "Shipment Purge"](#) on page 228.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.

**Note:** Orders related to the shipments should have been purged by order purge. Shipments should have been closed by the Close Shipment transaction. See [Section A.2.8, "Close Shipment"](#) on page 119.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–176 Shipment History Purge Attributes

Attribute	Value
Base Transaction ID	SHIPMENTHISTPRG
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–177 Shipment History Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Shipment History Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Removes qualifying records from the history tables listed under <a href="#">Tables Purged</a>.</li> <li>N- Test mode. Determines the rows that will be removed without actually removing them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–178 Shipment History Purge Statistics**

Statistic Name	Description
NumShipmentHistoriesPurged	Number of shipment histories purged.
NumShipmentLineHistoriesPurged	Number of shipment line histories purged.

## Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Shipment\_H table.

### Events Raised

None.

### Tables Purged

YFS\_ADDITIONAL\_ATTRIBUTE\_H

YFS\_ADDITIONAL\_DATE\_H

YFS\_CONTAINER\_DETAILS\_H

YFS\_CONTAINER\_STS\_AUDIT\_H

YFS\_INSTRUCTION\_DETAIL\_H

YFS\_SHIPMENT\_CONTAINER\_H

YFS\_SHIPMENT\_H

YFS\_SHIPMENT\_LINE\_H

YFS\_SHIPMENT\_LINE\_REQ\_TAG\_H

YFS\_SHIPMENT\_STATUS\_AUDIT\_H

YFS\_SHIPMENT\_TAG\_SERIAL\_H

YFS\_CONTAINER\_ACTIVITY\_H

### A.3.3.23 Shipment Purge

This purge removes shipment data from the system. This reduces load from the frequently accessed tables. This transaction picks up all shipments that have been marked as 'Closed' and archives them into their history tables. See [Section A.3.3.22, "Shipment History Purge"](#) on page 225. It also archives and purges the shipment's child tables.

This is not a pipeline transaction. It also does not work from the task queue. All orders in the shipment should have been purged.

The shipment should have been made before the lead time setup.

Any enterprise using the Yantra 7x Application Consoles must schedule purge transactions.



**Note:** Orders related to the shipments should have been purged by order purge. Shipments should have been closed by the Close Shipment transaction. See [Section A.2.8, "Close Shipment"](#) on page 119.

A shipment is picked up for purge if it meets the following criteria:

1. The shipment's last modification should fall before the lead time setup.
2. If the value of ShipmentClosedFlag field is set to "Y".
3. The order record should already be purged for all the shipment lines.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A-179 Shipment Purge Attributes**

Attribute	Value
Base Transaction ID	SHIPMENTPRG
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A-180 Shipment Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

**Table A–180 Shipment Purge Criteria Parameters**

Parameter	Description
EnterpriseCode	Optional. Enterprise for which the Shipment Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–181 Shipment Purge Statistics**

Statistic Name	Description
NumShipmentsPurged	Number of Shipments purged.
NumShipmentLinesPurged	Number of Shipment Lines purged.

### Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_Shipment table.

### Events Raised

None.

### Tables Purged

YFS\_ADDITIONAL\_ATTRIBUTES

YFS\_ADDITIONAL\_DATE

YFS\_CONTAINER\_DETAILS  
YFS\_LOAD\_SHIPMENT\_CHARGE  
YFS\_MONITOR\_ALERT  
YFS\_SHIPMENT\_CONTAINER  
YFS\_SHIPMENT\_STATUS\_AUDIT  
YFS\_SHIPMENT  
YFS\_INSTRUCTION\_DETAIL  
YFS\_SHIPMENT\_MONITOR\_ALERT  
YFS\_HEADER\_CHARGES  
YFS\_LINE\_CHARGES  
YFS\_TAX\_BREAKUP  
YFS\_SHIPMENT\_TAG\_SERIALS  
YFS\_SHIPMENT\_LINE  
YFS\_SHIPMENT\_LINE\_REQ\_TAG  
YFS\_ACTIVITY\_DEMAND  
YFS\_CONTAINER\_STS\_AUDIT  
YFS\_CONTAINER\_ACTIVITY

#### **A.3.3.24 Shipment Statistics Purge**

This transaction deletes the shipment statistics from the table older than the specified retention days.

This agent should be used whenever shipment statistics records need to be removed, such as after a application server restart.

#### **Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–182 Shipment Statistics Purge Attributes**

Attribute	Value
Base Transaction ID	PRG_SHIP_STATS
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–183 Shipment Statistics Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Shipment Statistics Purge needs to be run. If not passed, then all enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

## Statistics Parameters

The following are the statistics parameters for this transaction:

**Table A–184** *Shipment Statistics Purge Statistics*

Parameter	Description
NumShipmentStatisticsPurged	Number of shipment statistics purged.

## Pending Job Count

For this transaction the pending job count is the number of records that can be purged from the YFS\_SHIPMENT\_STATISTICS table.

## Events Raised

None.

## Tables Purged

YFS\_SHIPMENT\_STATISTICS

### A.3.3.25 Statistics Purge

This purge removes statistics data from the system. It purges all records older than the specified retention days.

---

**Note:** This purge only reads the rules defined by the hub. Enterprise overridden rules are not considered.

---



---

**Note:** Yantra recommends that this agent be run often. In a production environment, the YFS\_STATISTICS\_DETAIL table can grow very large, very quickly. It does not carry any old data, therefore it is a good practice to purge it aggressively, from once a day to once a week, depending on the table size.

---

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–185 Statistics Purge Attributes**

Attribute	Value
Base Transaction ID	STATTLPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–186 Statistics Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–187 Statistics Purge Statistics**

Statistic Name	Description
NumStatisticsPurged	Number of statistics purged

**Pending Job Count**

For this transaction the pending job count is the number of records that can be purged from the YFS\_STATISTICS\_DETAIL table.

**Events Raised**

None.

**Tables Purged**

YFS\_STATISTICS\_DETAIL

**A.3.3.26 Work Order History Purge**

This transaction deletes tasks previously archived by the Work Order Purge. See the [Section A.3.3.27, "Work Order Purge"](#) on page 237.

**Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–188 Work Order History Purge Attributes**

Attribute	Value
Base Transaction ID	WORK_ORDER_HISTORY_PURGE
Base Document Type	Work Order
Base Process Type	VAS
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

**Criteria Parameters**

The following are the criteria parameters for this transaction:

**Table A–189 Work Order History Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
Live	Optional. Mode in which to run. Defaults to N. <ul style="list-style-type: none"> <li>Y - Default value. Removes qualifying records from the history tables listed under <a href="#">Tables Purged</a>.</li> <li>N- Test mode. Determines the rows that will be removed without actually removing them.</li> </ul>
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
Node	Optional. Node for which the Work Order History Purge needs to be run. If not passed, then all nodes are monitored.
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that will only perform their tasks on the nodes with a matching node transactional velocity value.  Valid values are: LOW, HIGH, and any additional values defined by the Hub from Platform > System Administration > Agent Criteria Groups.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–190 Work Order History Purge Statistics**

Statistic Name	Description
NumWorkOrderHistoriesPurged	Number of work order histories purged.



**Pending Job Count**

For this transaction the pending job count is the number of records that can be purged from the YFS\_WORK\_ORDER\_H table.

**Events Raised**

None.

**Tables Purged**

YFS\_WO\_APPT\_USER\_H  
 YFS\_WORK\_ORDER\_H  
 YFS\_WORK\_ORDER\_APPT\_H  
 YFS\_WORK\_ORDER\_ACTIVITY\_H  
 YFS\_WORK\_ORDER\_ACTY\_DTL\_H  
 YFS\_WORK\_ORDER\_AUDT\_DTL\_H  
 YFS\_WORK\_ORDER\_COMPONENT\_H  
 YFS\_WORK\_ORDER\_COMP\_TAG\_H  
 YFS\_WORK\_ORDER\_HOLD\_TYPE\_H  
 YFS\_WORK\_ORDER\_HOLD\_TYPE\_LOG\_H  
 YFS\_WORK\_ORDER\_PROD\_DEL\_H  
 YFS\_WORK\_ORDER\_SERVICE\_LINE\_H  
 YFS\_WORK\_ORDER\_STS\_AUDIT\_H  
 YFS\_WORK\_ORDER\_TAG\_H

**A.3.3.27 Work Order Purge**

This time-triggered transaction purges all the work orders for a period greater than the retention days specified in the Work Order Purge criteria and those, which are either in the status of cancelled or completed.

**Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–191 Work Order Purge Attributes**

Attribute	Value
Base Transaction ID	WORK_ORDER_PURGE
Base Document Type	Work Order
Base Process Type	VAS
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–192 Work Order Purge Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
Live	Optional. Mode in which to run. Defaults to Y. <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under <a href="#">Tables Purged</a> to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that will be moved to history tables without actually moving them.</li> </ul>
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

**Table A–192 Work Order Purge Criteria Parameters**

Parameter	Description
Node	Optional. Node for which the Work Order Purge needs to be run. If not passed, then all nodes are monitored.
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that will only perform their tasks on the nodes with a matching node transactional velocity value.  Valid values are: LOW, HIGH, and any additional values defined by the Hub from Platform > System Administration > Agent Criteria Groups.

**Statistics Tracked**

The following statistics are tracked for this transaction:

**Table A–193 Work Order Purge Statistics**

Statistic Name	Description
NumWorkOrdersPurged	Number of work orders purged.

**Pending Job Count**

For this transaction the pending job count is the number of records that can be purged from the YFS\_WORK\_ORDER table.

**Events Raised**

None.

**Tables Purged**

YFS\_WO\_APPT\_USER

YFS\_WORK\_ORDER

YFS\_WORK\_ORDER\_ACTIVITY

YFS\_WORK\_ORDER\_ACTY\_DTL

YFS\_WORK\_ORDER\_HOLD\_TYPE

YFS\_WORK\_ORDER\_HOLD\_TYPE\_LOG  
YFS\_WORK\_ORDER\_APPT  
YFS\_WORK\_ORDER\_AUDT\_DTL  
YFS\_WORK\_ORDER\_COMPONENT  
YFS\_WORK\_ORDER\_COMP\_TAG  
YFS\_WORK\_ORDER\_PROD\_DEL  
YFS\_WORK\_ORDER\_SERVICE\_LINE  
YFS\_WORK\_ORDER\_STS\_AUDIT  
YFS\_WORK\_ORDER\_TAG

### A.4 Task Queue Syncher Time-Triggered Transactions

Many transactions use the task queue as their work repository. The workflow manager automatically creates tasks for transactions to handle the next processing step, as configured in your pipeline.

In some situations, the task queue repository may become out of date. For example, when reconfiguring the processing pipeline while the pipeline is active, the queue may go out of synch with the new pipeline configuration.

Alerts that indicate a halt in the lifecycle of a business document may indicate an out-dated task queue repository.

The task queue syncher transactions are designed to update the task queue repository with the latest list of open tasks to be performed by each transaction, based on the latest pipeline configuration.

The available task queue synchronizers are:

- [Load Execution Task Queue Syncher](#)
- [Order Delivery Task Queue Syncher](#)
- [Order Fulfillment Task Queue Syncher](#)
- [Order Negotiation Task Queue Syncher](#)

**Note:** Some of the statistics collected and tracked in Release 7.5 SP1 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Yantra.

### A.4.1 Load Execution Task Queue Syncher

This transaction synchronizes the task queue for the load execution process type.

#### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–194 Load Execution Task Queue Syncher Attributes**

Attribute	Value
Base Transaction ID	TASK_QUEUE_SYNCER_L_D
Base Document Type	Load
Base Process Type	Load Execution
Abstract Transaction	No
APIs Called	None

#### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–195 Load Execution Task Queue Syncher Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

#### Statistics Tracked

The following statistics are tracked for this transaction:

*Table A–196 Load Execution Task Queue Syncher Statistics*

Statistic Name	Description
NumTasksCreated	Number of tasks created.

**Pending Job Count**

None.

**Events Raised**

None.

**A.4.2 Order Delivery Task Queue Syncher**

This transaction synchronizes the order delivery process type.

**Attributes**

The following are the attributes for this time-triggered transaction:

*Table A–197 Order Delivery Task Queue Syncher Attributes*

Attribute	Value
Base Transaction ID	TASK_QUEUE_SYNCER_O_D
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	No
APIs Called	None

**Criteria Parameters**

The following are the criteria parameters for this transaction:

**Table A–198 Order Delivery Task Queue Syncher Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

**Statistics Tracked**

The following statistics are tracked for this transaction:

**Table A–199 Order Delivery Task Queue Syncher Statistics**

Statistic Name	Description
NumTasksCreated	Number of tasks created.

**Pending Job Count**

None.

**Events Raised**

None.

**A.4.3 Order Fulfillment Task Queue Syncher**

This transaction synchronizes the order fulfillment process type.

**Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–200 Order Fulfillment Task Queue Syncher Attributes**

Attribute	Value
Base Transaction ID	TASK_QUEUE_SYNCER_O_F
Base Document Type	Order
Base Process Type	Order Fulfillment

*Table A–200 Order Fulfillment Task Queue Syncher Attributes*

Attribute	Value
Abstract Transaction	No
APIs Called	None

**Criteria Parameters**

The following are the criteria parameters for this transaction:

*Table A–201 Order Fulfillment Task Queue Syncher Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

**Statistics Tracked**

The following statistics are tracked for this transaction:

*Table A–202 Order Fulfillment Task Queue Syncher Statistics*

Statistic Name	Description
NumTasksCreated	Number of tasks created.

**Pending Job Count**

None.

**Events Raised**

None.

**A.4.4 Order Negotiation Task Queue Syncher**

This transaction synchronizes the order negotiation process type.

**Attributes**

The following are the attributes for this time-triggered transaction:



**Table A–203 Order Negotiation Task Queue Syncher Attributes**

Attribute	Value
Base Transaction ID	TASK_QUEUE_SYNCER_O_N
Base Document Type	Order
Base Process Type	Order Negotiation
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this transaction:

**Table A–204 Order Negotiation Task Queue Syncher Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–205 Order Negotiation Task Queue Syncher Statistics**

Statistic Name	Description
NumTasksCreated	Number of tasks created.

### Pending Job Count

None.

### Events Raised

None.

## A.5 Monitors

Monitors are transactions that watch for processes or circumstances that are out of bounds and then raise alerts.

**Note:** Some of the statistics collected and tracked in Release 7.5 SP1 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Yantra.

**Note:** All Monitors with the exception of Negotiation and Old Order have a `CollectPendingJobs` criteria parameter. If this parameter is set to `N`, the agent will not collect information on the pending jobs for that monitor. This pending job information is used for monitoring the monitor in the *Yantra 7x System Management Guide*.

By default, `CollectPendingJobs` is set to `Y`. It can be helpful to set it to `N` if one monitor is performing a significant amount of `getPendingJobs` queries, and the overhead cost is too high.

### A.5.1 Availability Monitor

This time-triggered transaction monitors inventory availability. The Availability Monitor raises global alerts when the available inventory falls below the configured quantities on the current day, on subsequent days within the ATP time frame, and on subsequent days outside of the ATP time frame. The quantities for the days outside of the ATP time frame are determined by the maximum monitoring days. Unlike the schedule and release transactions, the Availability Monitor calculates the actual availability beyond the ATP horizon and does not assume infinite inventory.

#### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–206 Availability Monitor Attributes**

Attribute	Value
Base Transaction ID	ATP_MONITOR
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–207 Availability Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
MonitorOption	Optional. Specifies how to monitor inventory. Valid values are: <ul style="list-style-type: none"> <li>• 1 - current inventory</li> <li>• 0 - inventory within and outside of the ATP time frame. This is the default value.</li> </ul>
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
InventoryOrganizationCode	Optional. Valid owner inventory organization. Organization to process in this run. If not passed, all inventory organizations are processed.

### Statistics Tracked

None.

**Pending Job Count**

None.

**Events Raised**

No events are raised. Individual actions associated with the monitoring rule are executed.

Data published to the actions is `AVAILABILITY_MONITOR_dbd.txt`.

**A.5.2 Exception Monitor**

This time-triggered transaction monitors exceptions in your system as noted below. It monitors the exceptions logged in the system and escalates these exceptions:

- If an exception has not been assigned to a user by a certain time
- If an exception has not been resolved by a certain time
- If the active size of the queue is more than a certain maximum size

In order to prevent re-alerts on exceptions during every run of the Exception Monitor, specify a re-alert interval through Alert Management in the Yantra 7x Configurator. This attribute is associated with a queue and can be configured for each queue.

**Attributes**

The following are the attributes for this time-triggered transaction:

*Table A–208 Exception Monitor Attributes*

Attribute	Value
Base Transaction ID	EXCEPTION_MONITOR
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None

**Criteria Parameters**

The following are the criteria parameters for this monitor:

**Table A–209 Exception Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
QueueID	Optional. Defines the Alert Queue into which exceptions from this monitor are stored.
OrganizationCode	Optional. Organization to process in this run. If not passed, all inventory organizations are processed.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–210 Exception Monitor Statistics**

Statistic Name	Description
NumInboxProcessed	Number of alerts processed.
NumExceededQueueSizeAlerts	Number of actions raised when the number of unresolved alerts exceeds the queue's maximum active size.
NumUnResolvedAlerts	Number of actions raised when the unresolved alert time of an alert exceeds the queue's resolution time.
NumUnAssignedAlerts	Number of actions raised when the unassigned alert time of an alert exceeds the queue's assignment time.

### Pending Job Count

None.

### Events Raised

No events are raised. Individual actions associated with the monitoring rule are executed.

### A.5.3 Inventory Monitor

This time-triggered transaction monitors inventory availability at ship node level. It raises alerts at the ship node level when the available inventory exceeds or drops below the configured quantities.

This monitor uses the OPEN\_ORDER demand type to calculate available inventory at a given node. All supplies assigned to a supply type that is considered by the OPEN\_ORDER demand type are considered. For more information about configuring inventory supply and demand considerations, refer to the *Yantra 7x Inventory Synchronization Configuration Guide*.

#### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–211 Inventory Monitor Attributes**

Attribute	Value
Base Transaction ID	INVENTORY_MONITOR
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	checkAvailability()

#### Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–212 Inventory Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.

**Table A–212 Inventory Monitor Criteria Parameters**

Parameter	Description
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
InventoryOrganizationCode	Optional. Valid inventory owner organization. Organization to process in this run. If not passed, all inventory organizations are processed.

**Statistics Tracked**

None.

**Pending Job Count**

None.

**Events Raised**

No events are raised. Individual actions associated with the monitoring rule are executed.

Data published to the actions is <YFS\_HOME>/documentation/api\_javadocs/dbd/INVENTORY\_MONITOR\_dbd.txt.

**A.5.4 Negotiation Monitor**

This time-triggered transaction alerts the Enterprise when a negotiation remains in a particular status for a specific amount of time. This also monitors the negotiation expiration date. This time-triggered transaction invokes the actions configured against the negotiation statuses.

Configure status Expired (2000) to monitor negotiation expiration date.

Use this monitor in environments where Order or order release has to go through a negotiation phase and you want to monitor the negotiation.

**Attributes**

The following are the attributes for this time-triggered transaction:

**Table A–213** *Negotiation Monitor Attributes*

Attribute	Value
Base Transaction ID	ORD_NEGOTIATION_MONITOR
Base Document Type	Order
Base Process Type	Order Negotiation
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–214** *Negotiation Monitor Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Negotiation Monitor needs to be run. If not passed, then all enterprises are monitored.
Status	The negotiation status you are monitoring.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–215** *Negotiation Monitor Statistics*

Statistic Name	Description
NumNegotiationsProcessed	Number of negotiations processed.
NumNegotiationsRequiringAlert	Number of negotiations which have at least one alert raised.



### Pending Job Count

None.

### Events Raised

This invokes the actions configured against the negotiation statuses.

Key Data - Not Applicable.

Data Published - `YCP_getNegotiationDetails_output.xml`

## A.5.5 Order Monitor

This time-triggered transaction alerts the enterprise when an order remains in a particular status for a specific amount of time.

Use this monitor if you care to track how long orders stay in a particular state.

This transaction is deprecated for this release.

**Note:** The same relog interval is used for all document types.

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–216** *Order Monitor Attributes*

Attribute	Value
Base Transaction ID	ORDER_MONITOR
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–217 Order Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Order Monitor needs to be run. If not passed, then all enterprises are monitored.
Status	Optional. The order status you want to monitor (if not monitoring a status range).
LeastAge1	This field is not used in this version.
FromStatus	Optional. Statuses to monitor that are greater than or equal to the passed status (if not monitoring a specific status).
ToStatus	Optional. Statuses to monitor that are less than or equal to the passed status (if not monitoring a specific status).

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–218 Order Monitor Statistics**

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumOrdersRequiringAlert	Number of orders which have at least one alert raised.

### Pending Job Count

None.

### Events Raised

No events are raised. Individual actions associated with the monitoring rule are executed.

Data published to the actions is `ORDER_MONITOR_dbd.txt`.

### A.5.6 Enhanced Order Monitor

The enhanced order monitor allows you to monitor the following situations:

- Milestone x has not been reached y hours before a given date type.
- Milestone x has not been reached within y hours of a given date type.
- Milestone x has not been reached within y hours of milestone z.
- Milestone x has been reached y hours before a given date type.
- Milestone x has been reached within y hours of a given date type.
- Milestone x has been reached within y hours after milestone z.
- The order has been in status x for y hours.
- Date type x is y hours before date type z.
- Date type x is y hours after date type z.

The order monitor can be configured to monitor the following system date types for Sales Order and Purchase Order document types:

- Actual Order Date - Read from the `ORDER_DATE` column of the `YFS_ORDER_HEADER` table
- Requested Ship Date - If there is an order release, read from the `REQ_SHIP_DATE` column of the `YFS_ORDER_RELEASE` table. Otherwise, read from the `REQ_SHIP_DATE` of the `YFS_ORDER_LINE` table.
- Expected Ship Date - Read from the `EXPECTED_SHIPMENT_DATE` column of the `YFS_ORDER_LINE_SCHEDULE` table. If it is null, uses the same logic as Requested Ship Date.
- Actual Ship Date - If the date is before 01/01/2500, read from the `EXPECTED_SHIPMENT_DATE` column of the `YFS_ORDER_LINE_SCHEDULE` table. If the date is on or after 01/01/2500, this date type is returned as null.
- Requested Delivery Date - If there is a release, read from the `REQ_DELIVERY_DATE` column of the `YFS_ORDER_RELEASE` table. Otherwise, read from the `REQ_DELIVERY_DATE` of the `YFS_ORDER_LINE` table.

- Expected Delivery Date - Read from the EXPECTED\_DELIVERY\_DATE column of the YFS\_ORDER\_LINE\_SCHEDULE table. If it is null, uses the same logic as Requested Delivery Date.
- Actual Delivery Date - If the date is before 01/01/2500, read from the EXPECTED\_DELIVERY\_DATE column of the YFS\_ORDER\_LINE\_SCHEDULE table. If the date is on or after 01/01/2500, this date type is returned as null.

---

---

**Note:** For Order Fulfillment, Planned Order Execution, Reverse Logistics, and Purchase Order Execution pipelines, the system defined dates such as Shipment and Delivery are stored without a time component. Therefore when you configure a rule using these dates, all time computations are carried out assuming they are always 12:00:00 AM.

---

---

For more information about milestones, date types, and monitoring rules, refer to the *Yantra 7x Distributed Order Management Configuration Guide*, *Yantra 7x Supply Collaboration Configuration Guide*, and/or *Yantra 7x Reverse Logistics Configuration Guide*.

---

---

**Important:** If you run the Enhanced Order Monitor, you must configure and run the Close Order time-triggered transaction in all applicable pipelines. For more details on the Close Order time-triggered transaction, see [Section A.2.6, "Close Order"](#) on page 115.

---

---

---

---

**Note:** The same relog interval is used for all document types.

---

---

### Attributes

The following are the attributes for this time-triggered transaction:

**Table A–219 Enhanced Order Monitor Attributes**

Attribute	Value
Base Transaction ID	ORDER_MONITOR_EX
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	None

### Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–220 Enhanced Order Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Order Monitor needs to be run. If not passed, then all enterprises are monitored.
FromStatus	Optional. Statuses to monitor that are greater than or equal to the passed status.
ToStatus	Optional. Statuses to monitor that are less than or equal to the passed status.

### Statistics Tracked

The following statistics are tracked for this monitor:

**Table A–221 Enhanced Order Monitor Statistics**

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumAlertsRaised	Number of alerts raised.

Pending Job Count

For this transaction the pending job count is the number of open orders with the value of NEXT\_ALERT\_TS less than or equal to (<=) the current date.

Events Raised

Table A–222 Events Raised by the Enhanced Order Monitor Transaction

Transaction/Event	Key Data	Data Published*	Template Support?
ON_AUTO_CANCEL	ORDER_MONITOR_dbd.txt	YFS_ORDER_MONITOR_EX.ON_AUTO_CANCEL.html	Yes
* These files are located in the following directory: <YFS_HOME>/documentation/api_javadocs/XSD/HTML			

**Note:** The Enhance Order Monitor transaction raises the ON\_AUTO\_CANCEL event, but does not cancel the order. A service on this event should be configured to cancel the order.

A.5.7 Enhanced Return Monitor

The enhanced return monitor allows you to monitor the following situations:

- Milestone x has not been reached y hours before a given date type.
- Milestone x has not been reached within y hours of a given date type.
- Milestone x has not been reached within y hours of milestone z.
- Milestone x has been reached y hours before a given date type.
- Milestone x has been reached within y hours of a given date type.
- Milestone x has been reached within y hours after milestone z.
- The order has been in status x for y hours.
- Date type x is y hours before date type z.

- Date type x is y hours after date type z.

The enhanced return monitor can be configured to monitor the following system date types:

- Actual Order Date - Read from the ORDER\_DATE column of the YFS\_ORDER\_HEADER table
- Requested Ship Date - If there is an order release, read from the REQ\_SHIP\_DATE column of the YFS\_ORDER\_RELEASE table. Otherwise, read from the REQ\_SHIP\_DATE of the YFS\_ORDER\_LINE table.
- Expected Ship Date - Read from the EXPECTED\_SHIPMENT\_DATE column of the YFS\_ORDER\_LINE\_SCHEDULE table. If it is null, uses the same logic as Requested Ship Date.
- Actual Ship Date - If the date is before 01/01/2500, read from the EXPECTED\_SHIPMENT\_DATE column of the YFS\_ORDER\_LINE\_SCHEDULE table. If the date is on or after 01/01/2500, this date type is returned as null.
- Requested Delivery Date - If there is a release, read from the REQ\_DELIVERY\_DATE column of the YFS\_ORDER\_RELEASE table. Otherwise, read from the REQ\_DELIVERY\_DATE of the YFS\_ORDER\_LINE table.
- Expected Delivery Date - Read from the EXPECTED\_DELIVERY\_DATE column of the YFS\_ORDER\_LINE\_SCHEDULE table. If it is null, uses the same logic as Requested Delivery Date.
- Actual Delivery Date - If the date is before 01/01/2500, read from the EXPECTED\_DELIVERY\_DATE column of the YFS\_ORDER\_LINE\_SCHEDULE table. If the date is on or after 01/01/2500, this date type is returned as null.

**Note:** For Order Fulfillment, Planned Order Execution, Reverse Logistics, and Purchase Order Execution pipelines, the system defined dates such as Shipment and Delivery are stored without a time component. Therefore when you configure a rule using these dates, all time computations are carried out assuming they are always 12:00:00 AM.

For more information about milestones, date types, and monitoring rules, refer to the *Yantra 7x Distributed Order Management Configuration Guide*, *Yantra 7x Supply Collaboration Configuration Guide*, and/or *Yantra 7x Reverse Logistics Configuration Guide*.

**Important:** If you run the Enhanced Return Monitor, you must configure and run the Close Order time-triggered transaction in all applicable pipelines. For more details on the Close Order time-triggered transaction, see [Section A.2.6, "Close Order"](#) on page 115.

**Note:** The same relog interval is used for all document types.

Attributes

The following are the attributes for this time-triggered transaction:

Table A–223 Enhanced Order Monitor Attributes

Attribute	Value
Base Transaction ID	RETURN_MONITOR_EX
Base Document Type	Return Order
Base Process Type	Reverse Logistics
Abstract Transaction	No
APIs Called	None



## Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–224 Enhanced Order Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Order Monitor needs to be run. If not passed, then all enterprises are monitored.
FromStatus	Optional. Statuses to monitor that are greater than or equal to the passed status.
ToStatus	Optional. Statuses to monitor that are less than or equal to the passed status.

## Statistics Tracked

The following statistics are tracked for this monitor:

**Table A–225 Enhanced Order Monitor Statistics**

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumAlertsRaised	Number of alerts raised.

## Pending Job Count

For this transaction the pending job count is the number of open orders with the value of NEXT\_ALERT\_TS less than or equal to ( $\leq$ ) the current date.

## Events Raised

No events are raised. Individual actions associated with the monitoring rule are executed.

The data published is RETURN\_MONITOR\_EX.xml.

### A.5.8 Real-time Availability Monitor

The Real-time Availability Monitor time-triggered transaction monitors the inventory availability of inventory items. It can be configured to raise the `REALTIME_AVAILABILITY_CHANGE` event when the inventory level for a given item changes in between thresholds defined in the Yantra 7x Configurator, in the Inventory Synchronization module.

It can be run in three modes:

- Activity Based: Raises the event in real time every time an item goes above or below one of the thresholds.
- Quick Sync: Re-sends the most recently published inventory availability information.
- Full Sync: Monitors all of the items regardless of activity and publishes the inventory information for all of the items.

In all cases, the percentage of future inventory availability will be used for considering inventory availability at retrieval time. For more information on future inventory availability, refer to the *Yantra 7x Inventory Synchronization Configuration Guide*.

Inventory available at the current date will be considered as on-hand. The processing time in the ATP rules must be set to at least 1 day, else past due supply will be included as part of on-hand inventory. For more information on configuring ATP Rules, refer to the *Yantra 7x Inventory Synchronization Configuration Guide*.

Demand of type `OPEN_ORDER` will be used in getting the inventory availability picture.

If sourcing is maintained, the Real-time Availability Monitor can either monitor the total availability across nodes or the availability at individual nodes.

When monitoring the total availability across nodes, the Real-time Availability Monitor monitors all nodes in the default distribution group of the inventory organization.

When monitoring the availability at individual nodes, the Real-time Availability Monitor monitors all nodes in a specified distribution group. For more information on configuring distribution groups and node-level inventory monitoring, refer to the *Yantra 7x Inventory Synchronization Configuration Guide*.

Inventory items without an Availability Monitor rule, or with a rule that is disabled, will not be processed by this time-triggered transaction.

If configured, the Real-time Availability Monitor will also consider the on hand and future inventory availability safety factor during monitoring. For more information on inventory availability safety factors and the `findInventory()` API, refer to the *Yantra 7x Inventory Synchronization Configuration Guide*, and the *Yantra 7x Javadocs*.

When the on hand quantity is greater than the configured low threshold, the `REALTIME_ONHAND` alert type is raised, and the alert level is based on the on hand quantity.

When the on hand quantity falls below the configured low threshold, the `REALTIME_FUTURE_MAX` alert type is raised, and the alert level is based on the total future supply (`FutureAvailableQuantity`) with `FirstFutureAvailabeDate` set to the date on which the first future supply is available, and `FutureAvailableDate` set to the date on which the maximum future supply is available.

**Note:** When the Real-time Availability Monitor is run in activity based mode, changing one of the thresholds of an inventory item will not cause the agent to monitor it unless there is a change in activity. For example, if item I with available quantity 700 is being monitored with a low threshold of 600, and the low threshold is then changed to 1000, no event will be published unless there is change in I's activity. In order to ensure that in such a scenario I is not left unmonitored, call the `createInventoryActivity` API when changing a monitoring rule for an item.

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–226 Real-time Availability Monitor Attributes**

Attribute	Value
Base Transaction ID	REALTIME_ATP_MONITOR
Base Document Type	General
Base Process Type	General

**Table A–226 Real-time Availability Monitor Attributes**

Attribute	Value
Abstract Transaction	No
APIs Called	FindInventory

### Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–227 Real-time Availability Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
InventoryOrganizationCode	Inventory organization code to use when MonitorOption is passed as 3. The inventory organization has to be an enterprise.  If this is not passed, the monitor runs for all inventory organizations.
MonitorOption	1 - Activity Based (Monitor based on distinct inventory items in YFS_INVENTORY_ACTIVITY table).  2 – Quick Sync (Re-raise event to publish information in the YFS_INVENTORY_ALERT table).  3 – Full Sync (Monitor based on all inventory items maintained by the inventory organization provided. If no InventoryOrganizationCode is provided, all inventory item will be monitored).  If not provided default value is 1.

**Table A–227 Real-time Availability Monitor Criteria Parameters**

Parameter	Description
ItemStatuses	List of valid statuses of items to be processed. Statuses must be separated by a , for example 3000,2000. This will only be used when MonitorOption is passed as 2 or 3. If provided, only items with the matching statuses will be monitored.
FromAlertTimestamp	<p>This will only be used when MonitorOption is passed as 2. If provided, the agent will raise the REALTIME_AVAILABILITY_CHANGE event to re-publish inventory availability information which was published between the time that the agent started and FromAlertTimestamp.</p> <p>If not provided, all inventory availability information published before the time that the agent started will be re-published.</p>
AllowedOverriddenCriteria	<p>If set to Y, the overridden value for the agent criteria parameters can be provided at the command line while triggering the agent in the following format:</p> <pre>&lt;AgentCriteriaAttribute&gt; &lt;OverriddenValue&gt;</pre> <p>For more information on passing these attributes see scheduling time-triggered transaction in <i>Yantra 7x Installation Guide</i>.</p>
FromLastNumberOfHours	<p>This will only be used when MonitorOption is passed as 2 to calculate the FromAlertTimestamp parameter, if necessary.</p> <p>If the FromAlertTimestamp parameter is not provided, it is calculated as current timestamp minus FromLastNumberOfHours.</p>

Statistics Tracked

None.

Pending Job Count

None.

Events Raised

The following events are raised by this time-triggered transaction:

Table A–228 Events Raised by the Realtime Availability Monitor Transaction

Transaction/Event	Key Data	Data Published*	Template Support?
REALTIME_AVAILABILITY_CHANGE	None	YFS_REALTIME_ATP_MONITOR.REALTIME_AVAILABILITY_CHANGE.html	Yes
* These files are located in the following directory: <YFS_HOME>/documentation/api_javadocs/XSD/HTML			

**Note:** Although described as 'real-time', availability changes may not be triggered immediately as inventory changes occur if the agent has a backlog of messages to process. Furthermore, this monitor exists as a time-triggered transaction, and thus monitors availability of inventory items only when the monitor is triggered based on the configured runtime properties.

A.5.9 Shipment Monitor

This time-triggered transaction reports on the states of a shipment, based on rules in the YFS\_MONITOR\_RULE table. This transaction allows you to monitor the following situations:

- If the Shipment has been in a status for more than a specified amount of time
- If a specified date that is associated with the shipment is:

- n hours before another specified date
- n hours after another specified date
- n hours not before another specified date
- n hours not after another specified date

Monitoring rules can be configured for shipment's origin and destination points.

Monitoring rules cannot be configured for a shipment's intermediate pickup and drop off points. A shipment has intermediate pickup or drop off only if it has multiple pickup or drop off points. For example, a shipment has more than one loads carrying it. The shipment status on first load deposit, second load deposit, and so forth cannot be monitored. Once the last load deposits the shipment at its destination, then the shipment status can be marked and monitored.

This is not a pipeline transaction. It also does not work from the task queue.

For more information about milestones, date types, and monitoring rules, see the *Yantra 7x Distributed Order Management Configuration Guide*, *Yantra 7x Supply Collaboration Configuration Guide*, and/or *Yantra 7x Reverse Logistics Configuration Guide*.

## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–229 Shipment Monitor Attributes**

Attribute	Value
Base Transaction ID	SHIPMENT_MONITOR
Base Document Type	Order
Base Process Type	Order Delivery
Abstract Transaction	No
APIs Called	None

## Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–230 Shipment Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Shipment Monitor needs to be run. If not passed, then all enterprises are monitored.

### Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–231 Shipment Monitor Statistics**

Statistic Name	Description
NumShipmentsMonitored	Number of shipments monitored.

### Pending Job Count

For this transaction the pending job count is the number of open shipments with the value of NEXT\_ALERT\_TS less than or equal to ( $\leq$ ) the current date.

### Events Raised

This invokes the actions configured against shipment statuses.

Key Data - Not Applicable.

Data Published - SHIPMENT\_MONITOR.xml

## A.5.10 Work Order Monitor

This time-triggered transaction alerts the enterprise when a work order remains in a particular state for a specific amount of time.

Use this monitor if you care to track how long work orders stay in a particular state.



## Attributes

The following are the attributes for this time-triggered transaction:

**Table A–232 Work Order Monitor Attributes**

Attribute	Value
Base Transaction ID	WORK_ORDER_MONITOR
Base Document Type	Work Order
Base Process Type	VAS Process
Abstract Transaction	No

## Criteria Parameters

The following are the criteria parameters for this monitor:

**Table A–233 Work Order Monitor Criteria Parameters**

Parameter	Description
Action	Required. Triggers the transaction. If left blank it defaults to get the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero) it defaults to 5000.
EnterpriseCode	Optional. Enterprise for which the Work Order Monitor needs to be run. If not passed then all enterprises are monitored.
Node	Optional. Node for which the Work Order Monitor needs to be run. If not passed then all nodes are monitored.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table A–234 Work Order Monitor Statistics**

Statistic Name	Description
NumWorkOrdersMonitored	Number of work orders monitored.

### **Pending Job Count**

For this transaction the pending job count is the number of Work Orders that are monitored, where NEXT\_ALERT\_TS less than or equal to ( $\leq$ ) current date.

### **Events Raised**

No events are raised. Individual actions associated with the monitoring rule are executed. Data published to the actions is workOrder\_dbd.txt.

# B

## Condition Builder Attributes

---

Statements in the condition builder are built using attributes that are defined throughout the Yantra 7x Configurator. This appendix describes all of those attributes for each process type.

Click one of the links below to be taken to the appropriate condition builder attributes description.

### **Sales Order**

- [Order Fulfillment](#)
- [Order Negotiation](#)
- [Outbound Shipment](#)

### **Planned Order**

- [Planned Order Execution](#)
- [Planned Order Negotiation](#)

### **Return Order**

- [Reverse Logistics](#)
- [Return Shipment](#)
- [Return Receipt](#)

### **Template Order**

- [Template Order](#)

### **Purchase Order**

- [Purchase Order Execution](#)

- Purchase Order Negotiation
- Inbound Shipment
- Purchase Order Receipt

### **Transfer Order**

- Transfer Order Execution
- Transfer Order Delivery
- Transfer Order Receipt

### **Load**

- Load Execution

### **General**

- General
- WMS Putaway
- WMS Layout Definition
- WMS Inventory
- Trailer Loading
- Task Execution
- Move Request Execution
- Manifesting
- Over Pack Build

### **Count**

- Count Execution

### **Container**

- Pack Process

### **Wave**

- Outbound Picking

## Work Order

- [VAS Process](#)

# B.1 Sales Order

## B.1.1 Order Fulfillment

*Table B–1 Order Fulfillment Condition Builder Attributes*

Attribute	Description
<b>Order Attributes</b>	
Condition Variable 1	A variable that can be used for condition building. This is an existing field in the YFS_ORDER_LINE database table, and can be used to create conditions without extending the database.
Condition Variable 2	A variable that can be used for condition building. This is an existing field in the YFS_ORDER_LINE database table, and can be used to create conditions without extending the database.
Delivery Method	The delivery method of the order (shipment, pickup or delivery).
Disposition Code	The disposition code of the item. This field is only applicable for Reverse Logistics and Supply Collaboration.
Line Type	The type of the order line. Yantra 7x has no application logic associated with the order line type. This field can be set up as per your business practices.
Order Type	The type of the order. Yantra 7x has no application logic associated with the order type. This field can be set up as per your business practices.
Payment Status	The payment status of the order.
Sale Voided	The flag indicating whether the order is voided.
Transaction ID	The ID of the last transaction that was executed on the order.
<b>Participant Attributes</b>	
Bill To ID	The ID of the bill to address for the order.
Buyer Organization Code	The code of the organization that is buying the goods or services.

**Table B–1 Order Fulfillment Condition Builder Attributes**

Attribute	Description
Enterprise Code	The code of the enterprise on the order.
Receiving Node	The node that will be receiving the shipment for the order.
Seller Organization Code	The code of the organization that is selling the goods or services.
Ship Node	The node that will be shipping the shipment for the order.
Ship Node Interface Type	The interface type of the ship node on the order (External Application, Yantra 7x Application Consoles, Yantra Networked WMS, or Yantra WMS 6.2).
Ship To ID	The ID of the ship to address for the order.
Supplier Code	The code of the supplier for the order.
<b>Item Attributes</b>	
Item ID	The ID of the item on the order line.
Item Group Code	The group code of the service item. For example, if the service is a provided service item, then the item group code is PS.
Product Line	The product line of the item on the order line.
<b>Sourcing Attributes</b>	
Fulfillment Type	The fulfillment type of the order.
Intentional Backorder	The flag indicating whether the order was intentionally dropped into backordered status at order creation.
Is Firm Predefined Node	The flag indicating whether the node on the order is a firm predefined node.
Order Sourcing Classification	The order sourcing classification of the order.
Reservation Mandatory	The flag indicating whether the reservation is mandatory.
<b>Related Order Attributes</b>	
Chain Type	The chain type of the order.
Is Chained Line	The flag indicating whether the order line is chained with another order line.

**Table B–1 Order Fulfillment Condition Builder Attributes**

Attribute	Description
Is Derived Line	The flag indicating whether the order line is derived from another order line.
Order Purpose	The purpose of the order. If this is an exchange order, this field will be set to EXCHANGE.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.1.2 Order Negotiation

**Table B–2 Order Negotiation Condition Builder Attributes**

Attribute	Description
Enterprise Code	The code of the enterprise on the order.
Initiator Organization Code	The code of the organization that initiates the negotiation.
Negotiator Organization Code	The code of the organization that can accept, counter-offer, or reject the initiator's offer.
Negotiation Pipeline Key	The key of the negotiation pipeline this order is going through.
Negotiation Number	The negotiation number of this order.
Negotiation Rule Key	The key of the negotiation rule for this order.
Header Entity	The entity for which the negotiation was initiated. Currently, the only applicable entity is Order.
Negotiation Status	The status of the negotiation for this order.
Document Type	The document type for this order. Typical value is Sales Order.
Freight Terms	The freight terms for this order.

**Table B–2 Order Negotiation Condition Builder Attributes**

Attribute	Description
Payment Terms	The payment terms for this order.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

### B.1.3 Outbound Shipment

**Table B–3 Outbound Shipment Condition Builder Attributes**

Attribute	Description
Enterprise Code	The code of the enterprise on the outbound shipment.
Buyer Organization Code	The code of the organization that is buying the goods or services.
Seller Organization Code	The code of the organization that is selling the goods or services.
Ship Node	The node that will be shipping this shipment.
Ship Node Interface Type	The interface type of the ship node on the order (External Application, Yantra 7x Application Consoles, Yantra Networked WMS, or Yantra WMS 6.2).
Receiving Node	The node that will be receiving this shipment.
Ship Mode	The shipment mode that will be used for the shipment. For example, Parcel, Truck Load, Less-Than Truck Load.
Freight Terms	The freight terms for this shipment.
Carrier Type	The shipment's carrier type for this shipment.
Hazardous Materials Flag	The flag indicating whether these materials are hazardous.
ESP Check Required	The flag indicating whether an Economic Shipping Parameters check is required at shipment consolidation time.
Is Appointment Required	The flag indicating whether an appointment is required for a service execution.



**Table B–3 Outbound Shipment Condition Builder Attributes**

Attribute	Description
Routing Guide Maintained	The flag indicating whether a routing guide is maintained for this shipment.
Carrier	The carrier for the shipment.
Real-time Integration with Yantra WMS	The flag indicating whether the node this shipment is shipping from is integrating with Yantra 7x WMS. Setting this field to N means that you are integrating with Yantra DCS, or any other warehouse management system.
Manually Entered	The flag indicating whether or not the shipment was entered through the Yantra 7x Application Consoles.
Delivery Code	The code of the entity that pays for the transportation costs.
Country	The country that the shipment is being shipped to.
Delivery Method	The delivery method of the shipment (shipment, pickup or delivery).
Is Serial Requested	The flag indicating whether the shipment has any line with a specific serial number passed. If that is the case, a different outbound shipment process can be selected in the pipeline.
Is Provided Service	The flag indicating whether the shipment has an associated provided service item.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.2 Planned Order

### B.2.1 Planned Order Execution

The Planned Order Execution condition builder attributes are identical to the [Order Fulfillment](#) attributes.

## B.2.2 Planned Order Negotiation

The Planned Order Negotiation condition builder attributes are identical to the [Order Negotiation](#) attributes.

## B.3 Return Order

### B.3.1 Reverse Logistics

**Table B–4 Return Fulfillment Condition Builder Attributes**

Attribute	Description
<b>Order Attributes</b>	
Condition Variable 1	A variable that can be used for condition building. This is an existing field in the YFS_ORDER_LINE database table, and can be used to create conditions without extending the database.
Condition Variable 2	A variable that can be used for condition building. This is an existing field in the YFS_ORDER_LINE database table, and can be used to create conditions without extending the database.
Delivery Method	The delivery method of the return (shipment, pickup or delivery).
Disposition Code	The disposition code of the item.
Line Type	The type of the return line. Yantra 7x has no application logic associated with the return line type. This field can be set up as per your business practices.
Order Type	The type of the return. Yantra 7x has no application logic associated with the return type. This field can be set up as per your business practices.
Payment Status	The payment status of the return.
Sale Voided	The flag indicating whether the return is voided.
Transaction ID	The ID of the last transaction that was executed on the return.
<b>Participant Attributes</b>	
Bill To ID	The ID of the bill to address for the return.
Buyer Organization Code	The code of the organization that is buying the goods or services.

**Table B–4 Return Fulfillment Condition Builder Attributes**

Attribute	Description
Enterprise Code	The code of the enterprise on the return.
Receiving Node	The node that will be receiving the shipment for the return.
Seller Organization Code	The code of the organization that is selling the goods or services.
Ship Node	The node that will be shipping the shipment for the return.
Ship Node Interface Type	The interface type of the ship node on the return (External Application, Yantra 7x Application Consoles, Yantra Networked WMS, or YAntra WMS 6.2).
Ship To ID	The ID of the ship to address for the return.
Supplier Code	The code of the supplier for the return.
<b>Item Attributes</b>	
Item ID	The ID of the item on the return line.
Item Group Code	The group code of the service item. For example, if the service is a provided service item, then the item group code is PS.
Product Line	The product line of the item on the return line.
<b>Sourcing Attributes</b>	
Fulfillment Type	The fulfillment type of the return.
Intentional Backorder	The flag indicating whether the return was intentionally dropped into backordered status at return creation.
Is Firm Predefined Node	The flag indicating whether the node on the return is a firm predefined node.
Order Sourcing Classification	The order sourcing classification of the return.
Reservation Mandatory	The flag indicating whether the reservation is mandatory.
<b>Related Order Attributes</b>	
Chain Type	The chain type of the return.
Is Chained Line	The flag indicating whether the return line is chained with another return line.

**Table B–4 Return Fulfillment Condition Builder Attributes**

Attribute	Description
Is Derived Line	The flag indicating whether the return line is derived from another return line.
Order Purpose	This field is only applicable to sales orders.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

### B.3.2 Return Shipment

The Return Shipment condition builder attributes are identical to the [Outbound Shipment](#) attributes.

### B.3.3 Return Receipt

**Table B–5 Return Receipt Condition Builder Attributes**

Attribute	Description
Document Type	The document type on the receipt. Typical value is Return Order.
Enterprise Code	The code of the enterprise that owns the receipt.
Seller Organization Code	The code of the organization that is selling the goods or services.
Ship Node	The node where the shipment was shipped out of.
Buyer Organization Code	The code of the organization that is buying the goods or services.
Receiving Node	The node where the shipment was received.
Receiving Node Interface Type	The interface type of the receiving node on the order (External Application, Yantra 7x Application Consoles, Yantra Networked WMS, or Yantra WMS 6.2).
Ship Mode	The shipment mode that will be used for the shipment. For example, Parcel, Truck Load, Less-Than Truck Load.
Freight Terms	The freight terms on the receipt.

**Table B–5 Return Receipt Condition Builder Attributes**

Attribute	Description
Carrier Type	The carrier type on the receipt.
Is Hazardous Material	The flag indicating whether there are hazardous materials that are being received.
Is Inspection Pending	The flag indicating whether there is an inspection pending on this return.
Is Receiving Node Integrated Real Time	The flag indicating whether the receiving node is integrating with Yantra WMS, or with another WMS system.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.4 Template Order

The Template Order condition builder attributes are identical to the [Order Fulfillment](#) attributes.

## B.5 Purchase Order

### B.5.1 Purchase Order Execution

**Table B–6 Purchase Order Execution Condition Builder Attributes**

Attribute	Description
<b>Order Attributes</b>	
Condition Variable 1	A variable that can be used for condition building. This is an existing field in the YFS_ORDER_LINE database table, and can be used to create conditions without extending the database.
Condition Variable 2	A variable that can be used for condition building. This is an existing field in the YFS_ORDER_LINE database table, and can be used to create conditions without extending the database.

**Table B–6 Purchase Order Execution Condition Builder Attributes**

Attribute	Description
Delivery Method	The delivery method of the inbound order (shipment, pickup or delivery).
Disposition Code	The disposition code of the item.
Line Type	The type of the inbound order line. Yantra 7x has no application logic associated with the inbound order line type. This field can be set up as per your business practices.
Order Type	The type of the inbound order. Yantra 7x has no application logic associated with the inbound order type. This field can be set up as per your business practices.
Payment Status	The payment status of the inbound order.
Sale Voided	The flag indicating whether the inbound order is voided.
Transaction ID	The ID of the last transaction that was executed on the inbound order.
<b>Participant Attributes</b>	
Bill To ID	The ID of the bill to address for the inbound order.
Buyer Organization Code	The code of the organization that is buying the goods or services.
Enterprise Code	The code of the enterprise on the inbound order.
Receiving Node	The node that will be receiving the shipment for the inbound order.
Seller Organization Code	The code of the organization that is selling the goods or services.
Ship Node	The node that will be shipping the shipment for the inbound order.
Ship Node Interface Type	The interface type of the ship node on the inbound order (External Application, Yantra 7x Application Consoles, Yantra Networked WMS, or Yantra WMS 6.2).
Ship To ID	The ID of the ship to address for the inbound order.
Supplier Code	The code of the supplier for the inbound order.
<b>Item Attributes</b>	

**Table B–6 Purchase Order Execution Condition Builder Attributes**

Attribute	Description
Item ID	The ID of the item on the inbound order line.
Item Group Code	The group code of the service item. For example, if the service is a provided service item, then the item group code is PS.
Product Line	The product line of the item on the inbound order line.
<b>Sourcing Attributes</b>	
Fulfillment Type	The fulfillment type of the inbound order.
Intentional Backorder	The flag indicating whether the inbound order was intentionally dropped into backordered status at inbound order creation.
Is Firm Predefined Node	The flag indicating whether the node on the inbound order is a firm predefined node.
Order Sourcing Classification	The order sourcing classification of the inbound order.
Reservation Mandatory	The flag indicating whether the reservation is mandatory.
<b>Related Order Attributes</b>	
Chain Type	The chain type of the inbound order.
Is Chained Line	The flag indicating whether the inbound order line is chained with another inbound order line.
Is Derived Line	The flag indicating whether the inbound order line is derived from another inbound order line.
Order Purpose	This field is only applicable to sales orders.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.5.2 Purchase Order Negotiation

**Table B–7 Purchase Order Negotiation Condition Builder Attributes**

Attribute	Description
Enterprise Code	The code of the enterprise on the inbound order.
Initiator Organization Code	The code of the organization that initiates the negotiation.
Negotiator Organization Code	The code of the organization that can accept, counter-offer, or reject the initiator's offer.
Negotiation Pipeline Key	The key of the negotiation pipeline this inbound order is going through.
Negotiation Number	The negotiation number of this inbound order.
Negotiation Rule Key	The key of the negotiation rule for this inbound order.
Header Entity	The entity for which the negotiation was initiated. Currently, the only applicable entity is Order.
Negotiation Status	The status of the negotiation for this inbound order.
Document Type	The document type for this inbound order. Typical value is Purchase Order.
Freight Terms	The freight terms for this inbound order.
Payment Terms	The payment terms for this inbound order.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.5.3 Inbound Shipment

The Inbound Shipment condition builder attributes are identical to the [Outbound Shipment](#) attributes.

## B.5.4 Purchase Order Receipt

The Purchase Order Receipt condition builder attributes are identical to the [Return Receipt](#) attributes.



## B.6 Transfer Order

### B.6.1 Transfer Order Execution

The Transfer Order Execution condition builder attributes are identical to the [Order Fulfillment](#) attributes.

### B.6.2 Transfer Order Delivery

The Transfer Order Delivery condition builder attributes are identical to the [Outbound Shipment](#) attributes.

### B.6.3 Transfer Order Receipt

The Transfer Order Receipt condition builder attributes are identical to the [Return Receipt](#) attributes.

## B.7 Load Execution

**Table B–8 Load Execution Condition Builder Attributes**

Attribute	Description
Load Type	The type of the load document.
Enterprise Code	The code of the enterprise on the load document.
Owner Organization Code	The code of the organization that owns the load document.
Carrier	The carrier used to carry the load.
Carrier Service Code	The code of the carrier service used to carry the load.
Ship Mode	The shipment mode that will be used for the shipment. For example, Parcel, Truck Load, Less-Than Truck Load.
Hazardous Material	The flag indicating whether hazardous materials are being carried in this load.
Origin Node	The node where the load originated from.
Destination Node	The node where the load is being shipped to.

**Table B–8 Load Execution Condition Builder Attributes**

Attribute	Description
Multiple Load Stop	The flag indicating whether or not a shipment will go through multiple stops to load or unload additional shipments.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.8 General

**Table B–9 General Condition Builder Attributes**

Attribute	Description
Enterprise Code	The code of the enterprise.
Organization Code	The code of the organization.
Provider Organization Code	The code of the organization that provides the service.
Ship Node	The node that will be shipping this shipment.
Supply Type	The supply type associated with the inventory status. Typical values are Onhand, Held, etc.
Item ID	The ID of the item on the order line.
Unit Of Measure	The unit of measure of the item.
Product Class	The inventory classification of an item based on the product's characteristics. Typical values are FQ - First Quality, SQ - Second Quality, etc.
Inventory Status	The inventory sub classification of the product, based on the results of the inventory control processes within the warehouse. Typical values are Good - Good Inventory, Damaged - Damaged inventory, Qlty-Hold - Quality Hold, etc.
Adjustment Type	The type of inventory adjustment. Typical values are Cycle Count, Receipt, Picking, Packing, Shipping, etc.
Alert Type	The type of alert raised when an exception occurs.
Carrier	The carrier used to carry the shipment.

**Table B–9 General Condition Builder Attributes**

Attribute	Description
Task Type	The Task Type applicable to a task. Typical values are Receipt, QC, Count, Replenishment, Retrieval, Putaway, VAS, Pack, Shipping, and Picking.
Assigned To User ID	The ID of the user to whom the task is assigned.
Task Status	The Task Status within the pipeline that the task travels through. Typical values are Open, Suggested, In Progress, Held, Completed, Canceled, etc.
Document Type	The document type for this order. Typical values are Sales Order, Purchase Order, Transfer Order, and Return Order.
Activity Group ID	The identifier for the activity group.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.9 WMS Putaway

The WMS Putaway condition builder attributes are identical to the [General](#) attributes.

## B.10 WMS Layout Definition

The WMS Layout Definition condition builder attributes are identical to the [General](#) attributes.

## B.11 WMS Inventory

The WMS Layout Inventory condition builder attributes are identical to the [General](#) attributes.

## B.12 Trailer Loading

The Trailer Loading condition builder attributes are identical to the [General](#) attributes.

## B.13 Task Execution

The Task Execution condition builder attributes are identical to the [General](#) attributes.

## B.14 Move Request Execution

The Move Request Execution condition builder attributes are identical to the [General](#) attributes.

## B.15 Manifesting

The Manifesting condition builder attributes are identical to the [General](#) attributes.

## B.16 Over Pack Build

The Over Pack Build condition builder attributes are identical to the [General](#) attributes.

## B.17 Count Execution

**Table B–10 Count Execution Condition Builder Attributes**

Attribute	Description
Enterprise Code	The code of the enterprise for which the count request is created.
Request Type	The type of count requested.
Count Program Name	The name of the count program for which the count request is created.
Node Key	The node where the count request is processed.
Zone ID	The zone where the count must be performed.
Location Size Code	The capacity of the location where the count must be performed.
Is LPN Level	The flag indicating whether the count tasks will be performed at the LPN level.

**Table B–10 Count Execution Condition Builder Attributes**

Attribute	Description
Is Case Level	The flag indicating whether the count tasks will be performed at the case level.
Is Pallet Level	The flag indicating whether the count tasks will be performed at the pallet level.
Is Item Level	The flag indicating whether the count tasks will be performed at the item level.
Is Resolvable	The flag indicating whether variance can be resolved for this count result.
Product Class	The inventory classification of an item based on the product's characteristics. Typical values are FQ - First Quality, SQ - Second Quality, etc.
Unit Of Measure	The unit of measure of the item that was counted.
Item Classification 1	The first item classification attribute for determining the Count Strategy.
Item Classification 2	The second item classification attribute for determining the Count Strategy.
Item Classification 3	The third item classification attribute for determining the Count Strategy.
Has Variance	The flag indicating whether the count request has a variance.
Has Absolute Variance	The flag indicating whether the count request has an absolute variance.
Variance Quantity	The difference in quantity (+/-) between the count result and system quantity.
Absolute Variance Quantity	The absolute difference between the count result and system quantity.
Variance Value	The difference in cost/value (+/-) between the count result and system quantity.
Absolute Variance Value	The absolute difference in cost/value between the count result and system quantity.

**Table B–10 Count Execution Condition Builder Attributes**

Attribute	Description
Has Variance With Previous Count	The flag indicating whether the variance between the current count result and previous count results is to be displayed.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.18 Pack Process

**Table B–11 Pack Process Condition Builder Attributes**

Attribute	Description
<b>Node Attributes</b>	
Ship Node	The node that will be shipping this shipment.
Receiving Node	The node that will be receiving this shipment.
Ship from Ship Node Interface Type	The interface type of the ship node from which the shipment is shipped (External Application, Yantra 7x Application Consoles, Yantra Networked WMS, or Yantra WMS 6.2).
Ship from Supplier Code	The code of the supplier that is shipping the shipment.
Ship from DCM Integration Real Time	The flag indicating whether the node from which the shipment is shipped uses Yantra 7x WMS.
Ship from Country	The code of the country from which the shipment is being shipped.
Ship to Ship Node Interface Type	The interface type of the ship node to which the shipment is shipped (External Application, Yantra 7x Application Consoles, Yantra Networked WMS, or Yantra WMS 6.2).
Ship to Supplier Code	The code of the supplier to whom the shipment is being shipped.
Ship to DCM Integration Real Time	The flag indicating whether the node to which the shipment is shipped uses Yantra 7x WMS.

**Table B–11 Pack Process Condition Builder Attributes**

Attribute	Description
Ship to Country	The code of the country to which the shipment is being shipped.
<b>Organization Attributes</b>	
Enterprise Code	The code of the enterprise that owns the shipment.
Buyer Organization Code	The code of the organization that is buying the goods or services.
Seller Organization Code	The code of the organization that is selling the goods or services.
<b>Shipment Attributes</b>	
Ship Mode	The shipment mode that will be used for the shipment. For example, Parcel, Truck Load, Less-Than Truck Load.
Carrier	The carrier used to carry the shipment.
Freight Terms	The freight terms of the shipment.
Delivery Code	The code of the entity that pays for the transportation costs.
Pack And Hold	The flag indicating whether the shipment needs to be packed and put away for retrieval at a later date.
Shipment Container Count	The number of containers in the shipment.
Shipment Containerized Flag	The flag indicating the containerization state of the shipment. The values are: 01 - not containerized, 02 - containerization in progress and 03 - containerization completed.
<b>Container Attributes</b>	
Is Shipment Container	The flag indicating whether the container belongs to a shipment.
Is Load Container	The flag indicating whether the container is part of a load.
Is Inventory Pallet	The flag indicating whether the container is an inventory pallet.
Is Converted From LPN	The flag indicating whether the inventory container has been converted to a shipment container.
Is Serial Capture Pending	The flag indicating whether the serial capture is pending for the container.

**Table B–11 Pack Process Condition Builder Attributes**

Attribute	Description
Is Pack Process Complete	The flag indicating whether any more pack activities are pending for the container.
Is Product Placing Complete	The flag indicating whether placing the product into the container according to the system's suggestion has been completed.
Requires VAS	The flag indicating whether the container requires value added services.
Has Child Containers	The flag indicating whether a container is a parent container having other containers.
Number of Items	The number of items contained in the container.
Container Type	The attribute that specifies whether a shipment container is a case or pallet.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## B.19 Outbound Picking

**Table B–12 Outbound Picking Condition Builder Attributes**

Attribute	Description
Activity Group ID	The identifier for the activity group.
Shipment Group ID	The identifier for the shipment group.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>



## B.20 VAS Process

**Table B–13 VAS Process Condition Builder Attributes**

Attribute	Description
Enterprise Code	The code of the enterprise that owns the item or license plate.
Provider Organization Code	The code of the organization that provides the service.
Node Key	The node, where the work orders are executed.
Purpose	The purpose for the work order (ORDER / STOCK / SHIP)
Service Item Group Code	The code of the service item group (KIT/DKIT/COMPL/INVC/PS)
Service Item ID	The identifier for the service Item.
Segment Type	The type of segment. This may be MTO (made to order) or MTC (made to customer).
Segment	The segment to which the inventory involved in the work order belongs.
Has Components	The flag indicating whether the work order has component items.
Status	The status of the work order.
Pre Call Status	The flag indicating the status of the pre-call process.
Appt Status	The status of the appointment. This will be in sync with the service order line. The appointment status is used in case of provided service work order.
Number Of Attempts	The number of attempts made to execute the work order.
Number Of Hours until Appointment	The number of hours left before the appointment for the service item.
Number Of Hours After Appointment	The number of hours after the last appointment for the service item.
Number Of Hours After Last Execution	The number of hours after the last attempt to execute the service.
Last Execution Success	The flag indicating whether the last attempt to execute the service was successful or not.

Table B–13 VAS Process Condition Builder Attributes

Attribute	Description
Open Work Order Flag	The flag indicating whether the execution of the work order has ended or not.
{ Enter Your Own Attribute }	<p>A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i>.</p> <p><b>Note:</b> This field is limited only to unexposed key attributes that are pre-defined by Yantra 7x as opposed to any XML attribute that you can enter.</p>

## A

---

actions, 48, 57  
    creation, 58  
activity codes  
    transportation  
        creating, 39  
        deleting, 41  
        modifying, 40  
Advanced Transit Days field, 32  
application rules side panel, 10

## B

---

base repositories, 48  
Break Bulk Node field, 97  
Business models, 2  
Buyer field, 86

## C

---

Carrier, 32  
Carrier Service Code field, 32, 97  
carrier service codes  
    creating, 31  
    deleting, 33  
    modifying, 33  
Carrier Service Description field, 32  
carrier special service codes  
    creating, 36  
    defining, 35  
    deleting, 38  
    modifying, 37

Carrier Special Services Code field, 37  
Carrier Special Services Description field, 37  
Carrier/Service field, 97  
charge categories  
    adding charge names, 65  
    creating, 63  
    deleting, 67  
    deleting charge names, 66  
    modifying, 66  
    modifying charge names, 66  
charge definitions, 63  
collect, 83  
common codes, 59  
condition builder, 271 to 293  
conditions, 48, 56  
    creating, 56  
configuration screens  
    accessing, 11  
Configurator  
    actions, 24  
        document types, 25  
        entering dates/times, 28  
        lists, 27  
        lookup functionality, 24  
        on-line help, 29  
        special characters, 29  
        troubleshooting, 29  
        users, 27  
    layout, 8  
    starting, 7  
    work area, 19  
Consider Buyer's Routing Guide field, 85  
Consolidate upto volume threshold of field, 89  
Consolidate upto weight threshold of field, 89

Consolidator field, 97, 98  
Country field, 96  
Customer PO # field, 88

## D

---

Delay shipment by not more than field, 89  
Delivery plans, 3  
DELIVERYPLANPRG purge rule, 69  
Department Code field, 89  
Description, 37  
Description field, 70  
Destinations, 4  
Distance Per Day field, 32  
Distributed Order Management, 1  
drop statuses, 54

## E

---

Economic Shipping Parameters, 86  
Economic shipping parameters maintained field, 89  
Effective Date field, 91  
events, 57

## F

---

Freight Term field, 85  
freight terms, 83  
    collect, 83  
    creating, 84  
    deleting, 86  
    modifying, 86  
    prepaid, 83  
    third party collect, 84  
    third party prepaid, 84  
Freight Terms field, 92  
From field, 97  
fulfillment, 48

## I

---

inheritance  
    determining, 11  
instruction types, 59 to 60

    creating, 59  
    deleting, 60  
    modifying, 60  
Inventory Synchronization, 1  
Item Classification field, 92

## L

---

load types  
    creating, 73  
    deleting, 74  
    modifying, 74  
LOADPRG purge rule, 69  
loads, 4  
Log File Name field, 71  
logistic attributes  
    freight terms  
        creating, 84  
        deleting, 86  
        modifying, 86  
Logistics Attributes branch, 83  
Logistics branch, 83  
Logistics Management, 1  
Long Description field, 85

## M

---

Maintained Externally field, 90  
Maintained in Yantra field, 90  
Mark For field, 89  
marketplaces, 3  
modification reasons, 43 to 45  
    creating, 43  
    deleting, 45  
    modifying, 45  
modification rules  
    modifying, 81  
multi-divisional corporations, 2

## N

---

Name field, 91  
negotiation, 48  
Node field, 98

Not Maintained field, 90

## O

---

Order # field, 89  
organization levels, 12  
    rules, 13  
organization rules, 13  
    loading another organization's rules, 17  
    overriding, 14  
origins, 4  
Override Freight Terms field, 98  
Override Ship To field, 98

## P

---

pickup statuses, 54  
pipeline determination, 48  
pipelines, 48  
    creating, 50  
    process type, 48  
    purchase order receipts  
        viewing action details, 57  
        viewing condition details, 56  
        viewing details, 49  
        viewing status details, 54  
        viewing transaction details, 51  
Platform, 2  
prepaid, 83  
Priority field, 97  
process types, 47  
Product Management, 1  
Publish Date field, 91  
Purge Code field, 70  
purge rules, 69  
    modifying, 69  
purges, 69

## R

---

receipt, 48  
Region Schema For Routing field, 89  
repositories, 48  
Retention Days field, 70

Reverse Logistics, 1  
Rollback Segment field, 70  
routing guide lines, 92  
Routing Guide # field, 91  
Routing guides  
    VICS, 90  
routing guides, 90  
    creating, 90  
    fields, 89  
    modifying, 92

## S

---

services, 48  
shipment, 48  
shipment planning, 86  
Shipments, 4  
Shipper field, 86  
Short Description field, 85  
State field, 96  
statuses, 48, 53, 54  
    creating, 54  
stop types  
    modifying, 77  
stops, 4  
Store# field, 97, 98  
Supersedes Routing Guide # field, 91  
Supply Collaboration, 1

## T

---

third party collect, 84  
third party prepaid, 84  
Third-party logistics models, 2  
To field, 97  
transactions, 48, 51  
    creating, 52  
transposition  
    activity codes, 39  
transportation  
    activity codes, 40, 41  
    defining, 39

## **V**

---

VICS, 90

## **W**

---

Warehouse Management, 2

Write to Log File field, 71

## **Z**

---

Zip Code field, 96