



Platform Configuration Guide

Release 7.5 SP1

May 2006



Copyright Notice

Platform 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This documentation, the software it describes, and the information and know-how they contain constitute the unpublished, proprietary, confidential and valuable trade secret information of Yantra Corporation, its affiliated companies or its or their licensors, and may not be used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	xxi
Structure	xxi
Yantra 7x Documentation	xxiv
Conventions	xxv

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 Platform Configuration	3
1.2.1 Participant Modeling	4
1.2.2 Yantra 7x Process Modeling	4
1.2.3 Security	4
1.2.4 System Administration	5
1.2.5 Unit of Measure	5
1.2.6 Internationalization	5
1.2.7 Presentation	5
1.2.8 Communication	5
1.2.9 Nomenclature	6
1.2.10 Yantra 7x Queue Management	6
1.2.11 Region Definition	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 Participants

3.1	Creating and Modifying an Organization	31
3.1.1	Defining an Organization's Primary Information	33
3.1.2	Assigning the Organization's Roles and Participant Associations	36
3.1.2.1	Defining Enterprise Attributes	40
3.1.2.2	Defining Seller Attributes.....	45
3.1.2.3	Defining Buyer Attributes	48
3.1.2.4	Defining Carrier Attributes	73
3.1.2.5	Defining Node Attributes	83
3.1.2.6	Defining a Node's Advanced Inventory Attributes.....	104
3.1.2.7	Defining an Organization's Advanced Inventory Attributes.....	106

3.1.2.8	Defining an Organization's Advanced Catalog Attributes.....	109
3.1.2.9	Defining an Organization's Advanced Capacity Attributes	111
3.1.3	Defining Communication Protocols	113
3.1.3.1	Creating a Communication Protocol	113
3.1.3.2	Modifying a Communication Protocol	114
3.1.3.3	Deleting a Communication Protocol.....	115
3.1.4	Defining an Organization's Payment Information.....	115
3.1.5	Viewing an Organization's Child Organizations.....	117
3.1.6	Defining an Organization's Calendars	118
3.1.6.1	Defining a Calendar's Defaults.....	122
3.1.6.2	Creating an Exception for a Particular Calendar Day	123
3.1.7	Viewing an Organization's Departments.....	125
3.1.7.1	Creating a Department.....	125
3.1.7.2	Modifying a Department	126
3.1.7.3	Deleting a Department.....	126
3.1.8	Modifying an Organization	127
3.2	Creating and Modifying an Organizational Hierarchy.....	127
3.2.1	Creating an Organizational Hierarchy	127
3.2.1.1	Adding an Organization to the Organizational Hierarchy.....	128
3.2.1.2	Removing an Organization from the Organizational Hierarchy.....	128
3.3	Creating Node Types	128
3.3.1	Creating a Node Type	129
3.3.2	Modifying a Node Type.....	129
3.3.3	Deleting a Node Type	130

4 Configuring Process Models

4.1	Document Type Configuration	131
4.1.1	The Process Modeling Tree	134
4.1.2	Creating a New Document Type	134
4.1.3	Modifying a Document Type's Description	136
4.1.4	Modifying a Process Type	137
4.1.4.1	Defining a Process Type's Primary Information	138
4.1.4.2	Defining a Process Type's Templates (Fulfillment Process Types Only)	145
4.2	Defining Process Type Pipelines.....	146

4.2.1	Defining Pipeline Determination.....	148
4.2.1.1	Condition Variables for Pipeline Determination.....	149
4.2.2	Creating a Pipeline	149
4.2.2.1	Condition Variables for Pipeline Setup	150
4.2.3	Saving a Pipeline as a Draft	150
4.2.4	Modifying a Pipeline	152
4.2.5	Defining a Pipeline's Monitoring Rules.....	153
4.2.6	Defining Transactions.....	158
4.2.6.1	Creating an Extended Transaction.....	160
4.2.6.2	Creating an Extended Transaction that is Derived from an Abstract Transaction	185
4.2.6.3	Modifying a Transaction.....	188
4.2.6.4	Deleting a Transaction	189
4.2.7	Defining Statuses	189
4.2.7.1	Creating an Extended Status.....	190
4.2.7.2	Modifying an Extended Status	191
4.2.7.3	Deleting an Extended Status.....	192
4.2.7.4	Defining Status Monitoring Rule Definitions	192
4.2.8	Defining Conditions	194
4.2.8.1	Creating a Condition	195
4.2.8.2	Modifying a Condition.....	198
4.2.8.3	Deleting a Condition	199
4.2.8.4	Viewing All Entities Affected by a Condition.....	199
4.2.9	Defining Actions	201
4.2.9.1	Creating an Action.....	202
4.2.9.2	Modifying an Action	207
4.2.9.3	Deleting an Action	207
4.2.10	Defining Service Definitions	208
4.2.10.1	Creating a Service	211
4.2.10.2	Saving a Service as a Draft.....	215
4.2.10.3	Saving a Service as Another Service	216

5 Configuring User Security

5.1	Defining User Groups.....	221
5.1.1	Creating and Modifying a User Group	222

5.1.1.1	Administering User Group Permissions	223
5.1.1.2	Viewing the Users Subscribed to a User Group	226
5.1.1.3	Setting Up Printer Preferences for a User Group	227
5.1.2	Deleting a User Group.....	230
5.2	Defining Users	230
5.2.1	Creating and Modifying a User	230
5.2.2	Setting Up Printer Preferences for a User.....	233
5.2.3	Deleting a User	236
5.3	Defining Data Security Groups	237
5.3.1	Creating a Data Security Group.....	238
5.3.2	Modifying a Data Security Group	240
5.3.3	Deleting a Data Security Group	240

6 Configuring System Administration Components

6.1	Defining Purge Criteria	241
6.1.1	Modifying a System Purge Criteria Rule	242
6.2	Defining User Exit Implementations	244
6.2.1	Defining a User Exit	245
6.3	Defining Installation Rules	248
6.4	Defining Agent Criteria Groups.....	254
6.4.1	Creating an Agent Criteria Group.....	254
6.4.2	Modifying an Agent Criteria Group	255
6.4.3	Deleting an Agent Criteria Group	256
6.5	Defining Initial Context Factory Codes.....	256
6.5.1	Creating an Initial Context Factory Code.....	256
6.5.2	Modifying an Initial Context Factory Code	257
6.5.3	Deleting an Initial Context Factory Code.....	258
6.6	Defining Health Monitor Rules	258
6.7	Viewing the List of Configured Servers	259
6.7.1	List of Sub Flows or Criteria ID Configured for Server	260

7 Configuring Units of Measure

7.1	Defining Quantity Units of Measure	261
7.1.1	Creating a Unit of Measure for Quantity.....	262
7.1.2	Modifying a Unit of Measure for Quantity	262

7.1.3	Creating a Unit of Measure Conversion Rate for Quantity	262
7.1.4	Modifying a Unit of Measure Conversion Rate for Quantity.....	263
7.1.5	Deleting a Unit of Measure Conversion Rate for Quantity	263
7.1.6	Deleting a Unit of Measure for Quantity	263
7.2	Defining Service Quantity Units of Measure	264
7.2.1	Creating a Unit of Measure for Service Quantity	264
7.2.2	Modifying a Unit of Measure for Service Quantity	264
7.2.3	Creating a Unit of Measure Conversion Rate for Service Quantity	265
7.2.4	Modifying a Unit of Measure Conversion Rate for Service Quantity.....	265
7.2.5	Deleting a Unit of Measure Conversion Rate for Service Quantity	265
7.2.6	Deleting a Unit of Measure for Service Quantity	266
7.3	Defining Dimension Units of Measure	266
7.3.1	Creating a Unit of Measure for Dimension	266
7.3.2	Modifying a Unit of Measure for Dimension.....	267
7.3.3	Creating a Unit of Measure Conversion Rate for Dimension.....	267
7.3.4	Modifying a Unit of Measure Conversion Rate for Dimension.....	268
7.3.5	Deleting a Unit of Measure Conversion Rate for Dimension.....	268
7.3.6	Deleting a Unit of Measure for Dimension	268
7.4	Defining Volume Units of Measure	269
7.4.1	Creating a Unit of Measure for Volume	269
7.4.2	Modifying a Unit of Measure for Volume.....	269
7.4.3	Creating a Unit of Measure Conversion Rate for Volume.....	270
7.4.4	Modifying a Unit of Measure Conversion Rate for Volume	270
7.4.5	Deleting a Unit of Measure Conversion Rate for Volume	270
7.4.6	Deleting a Unit of Measure for Volume	271
7.5	Defining Weight Units of Measure	271
7.5.1	Creating a Unit of Measure for Weight	271
7.5.2	Modifying a Unit of Measure for Weight	272
7.5.3	Creating a Unit of Measure Conversion Rate for Weight	272
7.5.4	Modifying a Unit of Measure Conversion Rate for Weight.....	272
7.5.5	Deleting a Unit of Measure Conversion Rate for Weight.....	273
7.5.6	Deleting a Unit of Measure for Weight	273
7.6	Defining Time Units of Measure	273
7.6.1	Creating a Unit of Measure for Time.....	274
7.6.2	Modifying a Unit of Measure for Time	274

7.6.3	Creating a Unit of Measure Conversion Rate for Time	274
7.6.4	Modifying a Unit of Measure Conversion Rate for Time.....	275
7.6.5	Deleting a Unit of Measure Conversion Rate for Time	275
7.6.6	Deleting a Unit of Measure for Time	275

8 Configuring Internationalization Rules

8.1	Defining Country Codes.....	277
8.1.1	Creating a Country Code Definition	278
8.1.2	Modifying a Country Code Definition	279
8.1.3	Deleting a Country Code Definition	279
8.2	Defining Language Codes	279
8.2.1	Creating a Language Definition	280
8.2.2	Modifying a Language Definition.....	280
8.2.3	Deleting a Language Definition.....	281
8.3	Defining Date Formats	281
8.3.1	Creating a Date Format.....	282
8.3.2	Modifying a Date Format	282
8.3.3	Deleting a Date Format	283
8.4	Defining Time Formats.....	283
8.4.1	Creating a Time Format	283
8.4.2	Modifying a Time Format.....	284
8.4.3	Deleting a Time Format.....	284
8.5	Defining Date/Time Formats	285
8.5.1	Creating a Date/Time Format.....	285
8.5.2	Modifying a Date/Time Format	286
8.5.3	Deleting a Date/Time Format	287
8.6	Defining Currency Definitions.....	287
8.6.1	Creating a Currency Definition	288
8.6.2	Modifying a Currency Definition.....	289
8.6.3	Deleting a Currency Definition.....	289
8.6.3.1	Defining the Default Reporting Conversion Rate and Euro Currency Code	289
8.7	Defining Currency Conversions.....	290
8.7.1	Creating a Currency Conversion	290
8.7.2	Modifying a Currency Conversion.....	291

8.7.3	Deleting a Currency Conversion.....	292
-------	-------------------------------------	-----

9 Configuring Presentation Components

9.1	Defining Locales	293
9.1.1	Creating a Locale	294
9.1.2	Modifying a Locale.....	296
9.1.3	Deleting a Locale.....	296
9.2	Defining Menus	297
9.2.1	Saving a Menu Group as a New Menu Grouping	297
9.2.2	Modifying Application Menu Details	298
9.2.3	Defining Menu Items.....	299
9.2.3.1	Creating a Menu Item	299
9.2.4	Modifying a Menu Item.....	301
9.2.5	Deleting a Menu Item	301
9.3	Defining Resources.....	302
9.3.1	Yantra 7x Resources	302
9.3.2	Application Consoles Entity	305
9.3.3	Application Consoles Related Entity.....	306
9.3.4	Application Consoles Search View	307
9.3.5	Application Consoles Detail View.....	308
9.3.6	Application Consoles List View.....	309
9.3.7	Application Consoles API	310
9.3.8	Application Consoles Inner Panel	313
9.3.9	Application Consoles Action	314
9.3.10	Application Consoles Link	316
9.3.11	Application Consoles Icon	317
9.4	Defining Themes	318
9.4.1	Creating a Theme.....	318
9.4.2	Modifying a Theme	319
9.4.3	Deleting a Theme	319
9.5	Defining Custom Common Code Types	320
9.5.1	Creating a Custom Common Code Type.....	320
9.5.2	Modifying a Custom Common Code Type	321
9.5.3	Deleting a Custom Common Code Type.....	322
9.6	Defining Custom Common Code Values	322

9.6.1	Adding Values to a Custom Common Code.....	322
9.6.2	Modifying a Custom Common Code's Values	323
9.6.3	Deleting a Custom Common Code's Values	324
9.7	Defining Custom Error Codes	324
9.7.1	Adding a Custom Error Code.....	325
9.7.2	Modifying a Custom Error Code	325
9.7.3	Deleting a Custom Error Code	326

10 Configuring Business Communication Components

10.1	Defining Protocol Codes	327
10.1.1	Creating a Protocol Code	328
10.1.2	Modifying a Protocol Code	328
10.1.3	Deleting a Protocol Code	328
10.2	Defining Document Format Codes.....	329
10.2.1	Creating a Document Format Code	329
10.2.2	Modifying a Document Format Code.....	330
10.2.3	Deleting a Document Format Code	330
10.3	Defining Business Document Codes	331
10.3.1	Creating a Business Document Code	331
10.3.2	Modifying a Business Document Code	332
10.3.3	Deleting a Business Document Code	332
10.3.4	Classifying an Existing Business Document	332

11 Configuring Nomenclature Components

11.1	Defining Nomenclature Codes	335
11.2	Defining Nomenclature Definitions	336
11.2.1	Creating Mappings between Nomenclature Definitions	338
11.3	Defining Nomenclature Configuration	340

12 Configuring Alert Queues

12.1	Creating an Alert Queue.....	345
12.1.1	Defining How Unresolved Alerts are Handled Based on Size.....	347
12.1.2	Defining How Unassigned Alerts are Handled Based on Time	348
12.1.3	Defining How Unresolved Alerts are Handled Based on Time.....	349

12.1.4	Viewing an Alert Queue's User List	350
12.2	Modifying an Alert Queue	350
12.3	Deleting an Alert Queue	350

13 Configuring Region Definitions

13.1	Defining Region Levels	353
13.1.1	Creating a Region Level	353
13.1.2	Modifying a Region Level	355
13.1.3	Deleting a Region Level	356
13.2	Defining Region Match Preferences	356
13.2.1	Setting a Region Match Preference	356
13.2.2	Deleting a Region Match Preference	357
13.3	Defining Region Schemas	357
13.3.1	Creating a Region Schema	357
13.3.1.1	Creating a Region	360
13.3.1.2	Modifying a Region	361
13.3.1.3	Deleting a Region	361
13.3.2	Modifying a Region Schema	361
13.3.3	Deleting a Region Schema	362

14 Configuring Devices

14.1	Defining a Device Type	363
14.1.1	Creating a Device Type	364
14.1.2	Modifying a Device Type	365
14.1.3	Deleting a Device Type	366
14.2	Defining a Device Sub Type	366
14.2.1	Creating a Device Sub Type	366
14.2.2	Modifying a Device Sub Type	367
14.2.3	Deleting a Device Sub Type	368
14.3	Defining a Device	368
14.3.1	Creating a Device	369
14.3.1.1	Setting Up a Device Attribute	372
14.3.2	Creating a New Device from a Device	373
14.3.3	Modifying a Device	375

14.3.4	Deleting a Device	376
--------	-------------------------	-----

15 Configuring Prints

15.1	Defining Print Documents	378
15.1.1	Creating a Print Document.....	378
15.1.2	Modifying a Print Document	380
15.1.3	Deleting a Print Document	380
15.2	Defining Label Formats	381
15.2.1	Creating a Label Format	381
15.2.2	Modifying a Label Format	383
15.2.3	Deleting a Label Format	384

A Time-Triggered Transaction Reference

A.1	Running Time-Triggered Transactions	386
A.2	Business Process Time-Triggered Transactions	387
A.2.1	Change Load Status	387
A.2.2	Change Shipment Status.....	389
A.2.3	Close Delivery Plan.....	390
A.2.4	Close Load	392
A.2.5	Close Manifest	394
A.2.6	Close Order	397
A.2.7	Close Receipts	399
A.2.8	Close Shipment.....	401
A.2.9	Collect Shipment Statistics	403
A.2.10	Complete Planned Order	405
A.2.11	Consolidate Additional Inventory	406
A.2.12	Consolidate To Shipment.....	408
A.2.13	Create Chained Order	411
A.2.14	Create Derived Order.....	413
A.2.15	Create Order Invoice	415
A.2.16	Create Shipment Invoice	417
A.2.17	ESP Evaluator	419
A.2.18	Mark Load as Trailer Loaded	421
A.2.19	Match Inventory.....	422
A.2.20	Payment Collection.....	424

A.2.21	Payment Execution	427
A.2.22	Post Inventory Match	430
A.2.23	Process Order Hold Type	432
A.2.24	Process Work Order Hold Type	434
A.2.25	Publish Negotiation Results	436
A.2.26	Release	438
A.2.27	Route Shipment	441
A.2.28	Schedule	443
A.2.29	Send Invoice	448
A.2.30	Send Order	450
A.2.31	Send Release	452
A.2.32	Start Order Negotiation	453
A.3	Time-Triggered Purge Transactions	455
A.3.1	Purge Strategy	456
A.3.2	Configuring Purge Transaction Log Files	456
A.3.3	Available Purges	457
A.3.3.1	Alert Purge	458
A.3.3.2	Capacity Purge	460
A.3.3.3	Delivery Plan Purge	463
A.3.3.4	Export Table Purge	465
A.3.3.5	Import Table Purge	467
A.3.3.6	Inventory Audit Purge	469
A.3.3.7	Inventory Purge	471
A.3.3.8	Inventory Supply Temp Purge	473
A.3.3.9	Load Purge	475
A.3.3.10	Manifest Purge	478
A.3.3.11	Negotiation History Purge	480
A.3.3.12	Negotiation Purge	482
A.3.3.13	Order History Purge	485
A.3.3.14	Order Purge	488
A.3.3.15	Order Status Audit Purge	493
A.3.3.16	Picklist Purge	495
A.3.3.17	Price List Purge	497
A.3.3.18	Receipt History Purge	499
A.3.3.19	Receipt Purge	501

A.3.3.20	Reprocess Error Purge	504
A.3.3.21	Reservation Purge	506
A.3.3.22	Shipment History Purge	507
A.3.3.23	Shipment Purge	510
A.3.3.24	Shipment Statistics Purge.....	513
A.3.3.25	Statistics Purge	515
A.3.3.26	Work Order History Purge.....	517
A.3.3.27	Work Order Purge	519
A.4	Task Queue Syncher Time-Triggered Transactions	522
A.4.1	Load Execution Task Queue Syncher	523
A.4.2	Order Delivery Task Queue Syncher.....	524
A.4.3	Order Fulfillment Task Queue Syncher.....	525
A.4.4	Order Negotiation Task Queue Syncher	526
A.5	Monitors	528
A.5.1	Availability Monitor	528
A.5.2	Exception Monitor	530
A.5.3	Inventory Monitor	532
A.5.4	Negotiation Monitor	533
A.5.5	Order Monitor	535
A.5.6	Enhanced Order Monitor.....	537
A.5.7	Enhanced Return Monitor	540
A.5.8	Real-time Availability Monitor.....	544
A.5.9	Shipment Monitor.....	548
A.5.10	Work Order Monitor	550

B Inventory and Capacity Change Transaction Reference

B.1	The Inventory Change Transaction	553
B.2	The Capacity Change Transaction	555

C Service Builder Nodes and Parameters

C.1	Transport Nodes	559
C.1.1	Database	560
C.1.1.1	Database Sender.....	560
C.1.1.2	Database Receiver.....	562
C.1.2	DCS 6.2 Database	565

C.1.2.1	DDCS 6.2 Database Sender	565
C.1.2.2	DCS 6.2 Database Receiver	566
C.1.3	Component Object Model (COM).....	568
C.1.4	Enterprise Java Beans (EJB)	569
C.1.5	File Input/Output (File I/O).....	571
C.1.5.1	File I/O Sender	571
C.1.5.2	File I/O Receiver	572
C.1.5.3	Enabling EOF Messages in the Platform Framework	575
C.1.6	File Transfer Protocol (FTP).....	578
C.1.6.1	FTP Sender.....	579
C.1.6.2	FTP Receiver.....	581
C.1.7	Hypertext Transport Protocol (HTTP).....	584
C.1.8	WebService	587
C.1.9	Synchronous WebLogic and MQSeries	589
C.1.10	WebLogic and MQSeries JMS Queue.....	593
C.1.10.1	WebLogic and MQSeries JMS Queue Sender	594
C.1.10.2	WebLogic and MQSeries JMS Receiver	599
C.1.11	Microsoft Message Queue (MSMQ)	603
C.1.11.1	MSMQ Sender	604
C.1.11.2	MSMQ Receiver	605
C.1.12	Receiver Link Exception Handling	607
C.2	Component Nodes	609
C.2.1	Alert.....	610
C.2.2	Application Programming Interface (API)	613
C.2.3	Composite Service.....	615
C.2.4	Condition	616
C.2.5	E-Mail.....	617
C.2.6	Nomenclature Runtime Component.....	619
C.2.6.1	Nomenclature Runtime.....	620
C.2.7	Router	620
C.2.8	Text Translator	621
C.2.9	XSL Translator	623
C.2.10	Defaulting Component	624
C.2.11	Data Security.....	626
C.2.12	Jasper Printer Component	628

C.3	Adapter Nodes	630
C.3.1	Sterling GIS	630
C.4	Installing the Service Builder Samples.....	631

D Text Translator Reference

D.1	Understanding File Formats	633
D.2	Text Translator Components	635
D.2.1	XSD Files	635
D.2.2	Schema Files	636
D.2.3	Defining a Schema File.....	637
D.3	Verifying the Text Translator Setup	637
D.3.1	Error Messages	638
D.4	Text Translator XSD Files	641
D.4.1	Text Translator Positional Receiver XSD File	641
D.4.2	Text Translator Delimited Receiver XSD File	648
D.4.3	Text Translator Positional Sender XSD File	651
D.4.4	Text Translator Delimited Sender XSD File	654
D.5	Running the Text Translator	656
D.5.1	Specifying the Text Translator Startup Script	656

E Document Types

F Condition Builder Attributes

F.1	Sales Order	663
F.1.1	Order Fulfillment	663
F.1.2	Order Negotiation.....	665
F.1.3	Outbound Shipment	666
F.2	Planned Order	667
F.2.1	Planned Order Execution	667
F.2.2	Planned Order Negotiation.....	668
F.3	Return Order	668
F.3.1	Reverse Logistics	668
F.3.2	Return Shipment.....	670
F.3.3	Return Receipt	670

F.4	Template Order.....	671
F.5	Purchase Order	671
F.5.1	Purchase Order Execution.....	671
F.5.2	Purchase Order Negotiation	674
F.5.3	Inbound Shipment.....	674
F.5.4	Purchase Order Receipt	674
F.6	Transfer Order	675
F.6.1	Transfer Order Execution.....	675
F.6.2	Transfer Order Delivery	675
F.6.3	Transfer Order Receipt.....	675
F.7	Load Execution	675
F.8	General.....	676
F.9	WMS Putaway.....	677
F.10	WMS Layout Definition.....	677
F.11	WMS Inventory.....	677
F.12	Trailer Loading.....	677
F.13	Task Execution.....	678
F.14	Move Request Execution.....	678
F.15	Manifesting.....	678
F.16	Over Pack Build.....	678
F.17	Count Execution.....	678
F.18	Pack Process.....	680
F.19	Outbound Picking	682
F.20	VAS Process	683

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, participants, users, and business processes as they pertain to their 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 Participants"

This chapter explains how you can configure participants in Yantra 7x. Each Participant is considered an organization with a defined role.

Chapter 4, "Configuring Process Models"

This chapter explains how you can configure process models. In Yantra 7x, Process Modeling is the set up of Yantra 7x business process workflow.

Chapter 5, "Configuring User Security"

This chapter explains how you can configure user security. Security must be set up to allow users access to the actions and views provided by the organization to which they belong.

Chapter 6, "Configuring System Administration Components"

This chapter explains how you can configure system level information including system level purge criteria, user exit implementations, and installation rules.

Chapter 7, "Configuring Units of Measure"

This chapter explains how you can define units of measure that enable you to set up standard units of measure to associate with your items and locales.

Chapter 8, "Configuring Internationalization Rules"

This chapter explains how you can configure internationalization rules that are used to set up rules and common codes associated with making Yantra 7x functional for international use.

Chapter 9, "Configuring Presentation Components"

This chapter explains how you can configure presentation components to provide an interface that enables you to customize the graphical user interface.

Chapter 10, "Configuring Business Communication Components"

This chapter explains how you can configure business communication components to define the codes and documents used to communicate between Yantra 7x and external systems as well as different business organizations within your business model.

Chapter 11, "Configuring Nomenclature Components"

This chapter explains how you can configure Nomenclature Runtime components to provide a mapping tool that allows you to configure unique terms you use to match unique terms your trading partners use.

Chapter 12, "Configuring Alert Queues"

This chapter explains how you can configure queue management to define rules and methods pertaining to user alert notifications.

Chapter 13, "Configuring Region Definitions"

This chapter explains how you can configure the components that are used by Yantra 7x geography engines. This individual components consisting of regions and region levels can be used to create region schemas that can then be used throughout Yantra 7x business application models whenever geography is considered.

Chapter 14, "Configuring Devices"

This chapter explains how you can configure hand-held and stationary devices that are used in a warehouse. These devices have their unique definitions and sometimes are associated specifically to stations or equipment.

Chapter 15, "Configuring Prints"

This chapter explains how you can configure prints. The operation of a warehouse requires numerous documents, be it labels or reports, to be printed daily. The printing of the documents is either initiated by the occurrence of specific events or is requested ad-hoc by a user.

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, "Inventory and Capacity Change Transaction Reference"

This chapter explains the inventory change transaction that is used for setting up events that involve inventory changes.

Appendix C, "Service Builder Nodes and Parameters"

This chapter provides a complete list of the transport, component and adapter nodes used in the Service Builder.

Appendix D, "Text Translator Reference"

This appendix describes the files you need to configure for the Text Translator node.

Appendix E, "Document Types"

This chapter provides a reference for the different document types used by Yantra 7x.

Appendix F, "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 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 in the Yantra 7x Configurator:

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

- 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 Platform Configuration

The Yantra 7x Platform is a collection of common components used across the application. These components are the infrastructure upon which all other business application modules in Yantra 7x are built.

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

- [Participant Modeling](#)
- [Yantra 7x Process Modeling](#)
- [Security](#)

- [System Administration](#)
- [Unit of Measure](#)
- [Internationalization](#)
- [Presentation](#)
- [Nomenclature](#)
- [Yantra 7x Queue Management](#)
- [Region Definition](#)

1.2.1 Participant Modeling

The Yantra 7x Configurator's Participant Modeling is used to create and maintain organizations and their relationships. Once an organization has been created, it is possible to define each organization's role, the services they provide and utilize, and the business elements required to support the collaborative processes between all of the participating organizations in the Hub. For more information about Participant Modeling, see [Chapter 3, "Configuring Participants"](#).

1.2.2 Yantra 7x Process Modeling

Yantra 7x Process Modeling defines process type details pipelines for business document types. A process type consists of one or more pipelines with the different statuses that a specific document type, such as a Sales Order, goes through during the lifecycle of that process type. For more information about Yantra 7x Process Modeling, see [Chapter 4, "Configuring Process Models"](#).

1.2.3 Security

Security must be set up to allow users access to the actions, views, and data within the user interface of the Yantra 7x Application Consoles and Yantra 7x Configurator. A user's access is limited to only those to which they have permission and data security rights.

Security is used to create users, user groups, and data security groups. For more information, see [Chapter 5, "Configuring User Security"](#).

1.2.4 System Administration

System Administration is used to configure system level information used throughout the Yantra 7x infrastructure. Through system administration system purge criteria, user exit implementations, and installation-time rules can be defined. For more information, see [Chapter 6, "Configuring System Administration Components"](#).

1.2.5 Unit of Measure

Unit of Measure is used to define standard units of measure to associate with your items and locales. With Yantra 7x provides unit of measure classifications for dimension, volume, weight, and time. In addition to defining new units of measure, Yantra 7x provides the ability to create conversion rates between different units of measure. For more information, see [Chapter 7, "Configuring Units of Measure"](#).

1.2.6 Internationalization

Internationalization provides a set of rules and common codes that can be used when implementing Yantra 7x in different international locales. Internationalization can be used to define country codes, language codes, date and time formats, and currency conversion rates. For more information, see [Chapter 8, "Configuring Internationalization Rules"](#).

1.2.7 Presentation

Presentation is used to configure locales, menus, themes, and resources necessary for user interface extensibility. For more information, see [Chapter 9, "Configuring Presentation Components"](#).

1.2.8 Communication

Communication is used to configure business communication components that define the codes and documents used to communicate between Yantra 7x and external systems as well as different business organizations within a business model. For more information, see [Chapter 10, "Configuring Business Communication Components"](#).

1.2.9 Nomenclature

Nomenclature provides components used to create a mapping of unique terms you use in your business model to a set of unique terms your trading partners use. For more information, see [Chapter 11, "Configuring Nomenclature Components"](#).

1.2.10 Yantra 7x Queue Management

Yantra 7x Queue Management is used to create queues for different users and types of alerts. These queues can be designed to notify specified users of alerts at configured levels and times. Yantra 7x Queue Management is also used to define how the configured users are notified. For more information, see [Chapter 12, "Configuring Alert Queues"](#).

1.2.11 Region Definition

Region Definition provides the components that are used by the Yantra 7x geography engine. The individual components, consisting of regions and region levels, can be used to create region schemas that can then be used throughout the Yantra 7x business application models whenever geography is considered. For more information about Region Definition, see [Chapter 13, "Configuring Region Definitions"](#).

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.

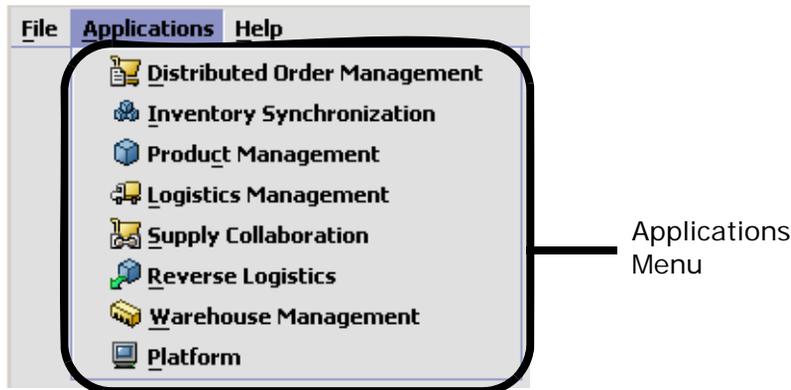
Note: Additionally, enterprise users who maintain an enterprise can access the Yantra 7x Configurator by means of `http://<Yantra 7x installation server>/yantra/console/login.jsp`.

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



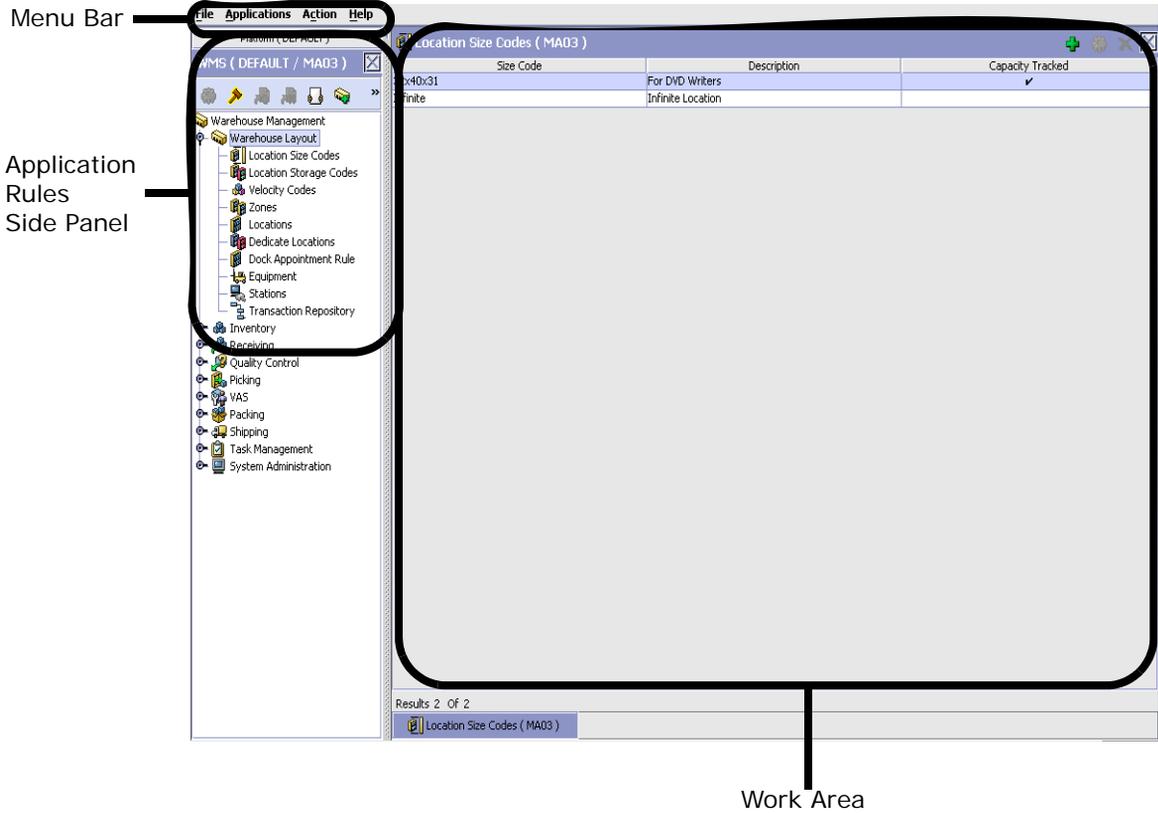
Each application focuses on a particular aspect of Yantra 7x and contains all of the rules, common codes, and settings necessary for Yantra 7x to work in a real-world business setting.

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.

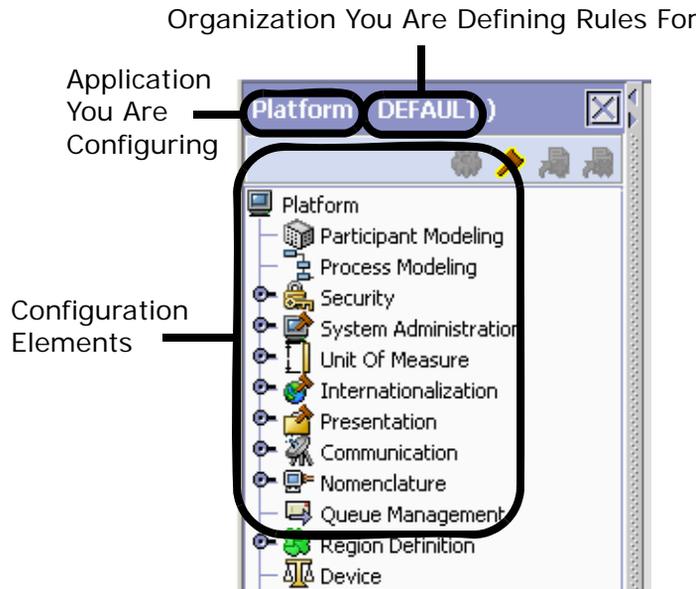
Figure 2–2 The Standard Configurator Application Interface



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.

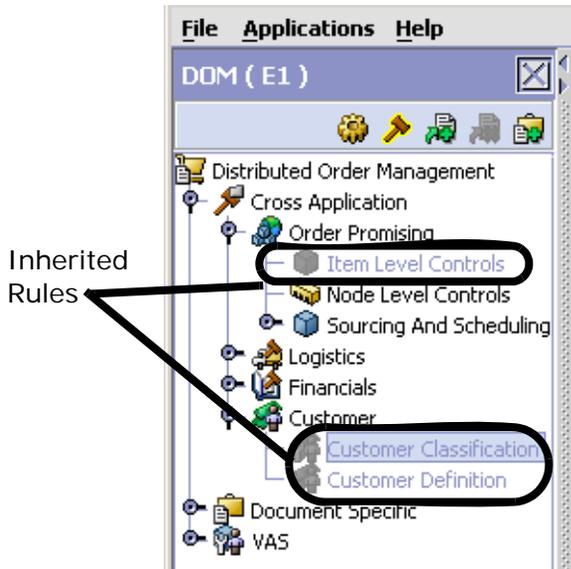
Table 2–1 Organization Level Rules

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.

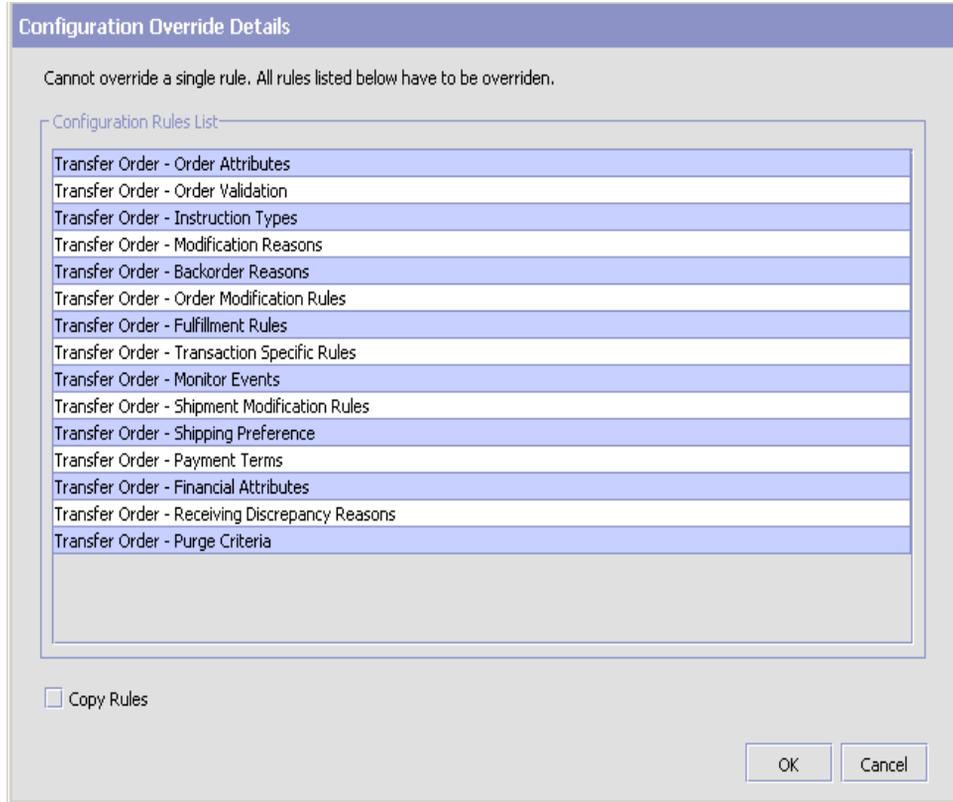
Figure 2–5 Override Configuration Icon

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.

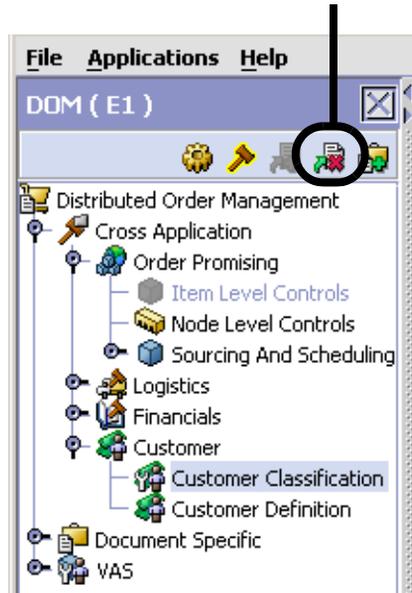
Figure 2–6 Example of Configuration Override Details Pop-Up Window



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

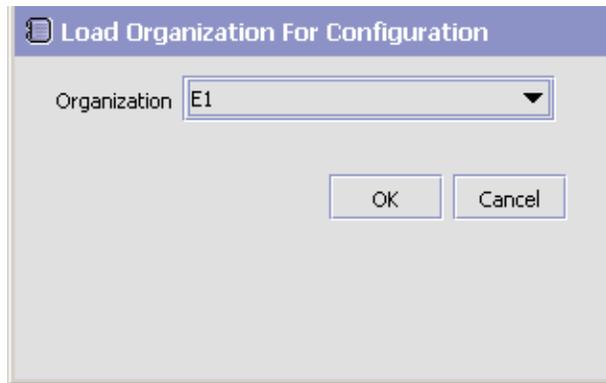
An administrator organization is set for every organization defined within the system. Only the administrator organization can modify the rules defined for a particular organization. If a particular organization administers multiple organizations, then they can load the rules of

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

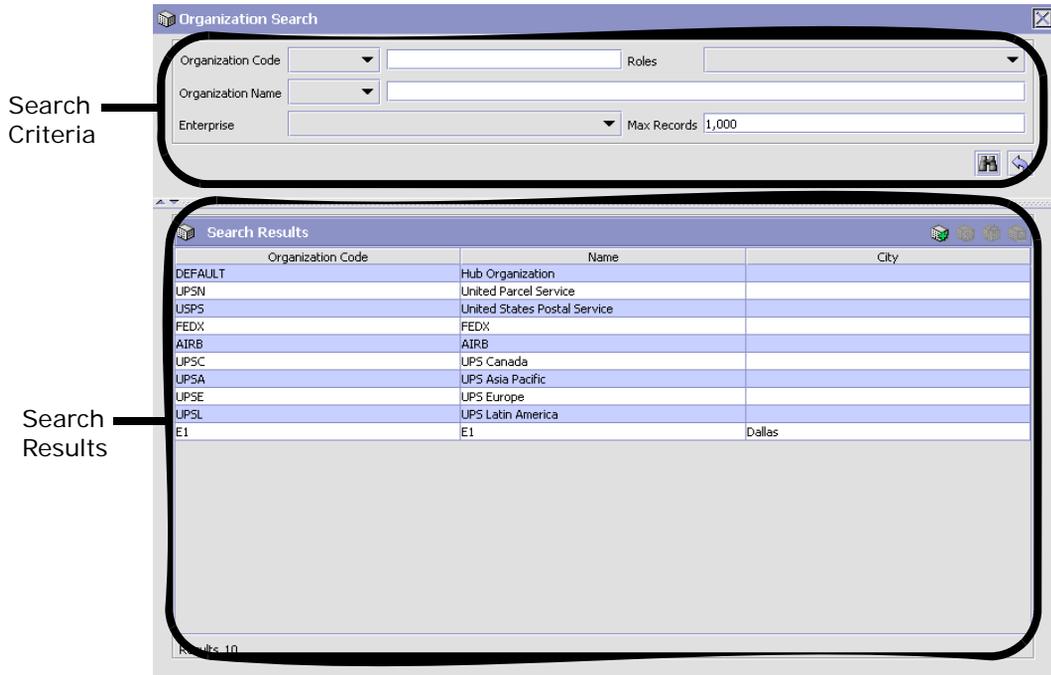
The work area is the main area in which different configuration screens appear. The following are the main types of screens that you will come across:

- [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


The screenshot shows a window titled "Dimension UOMs (DEFAULT)" with a table containing the following data:

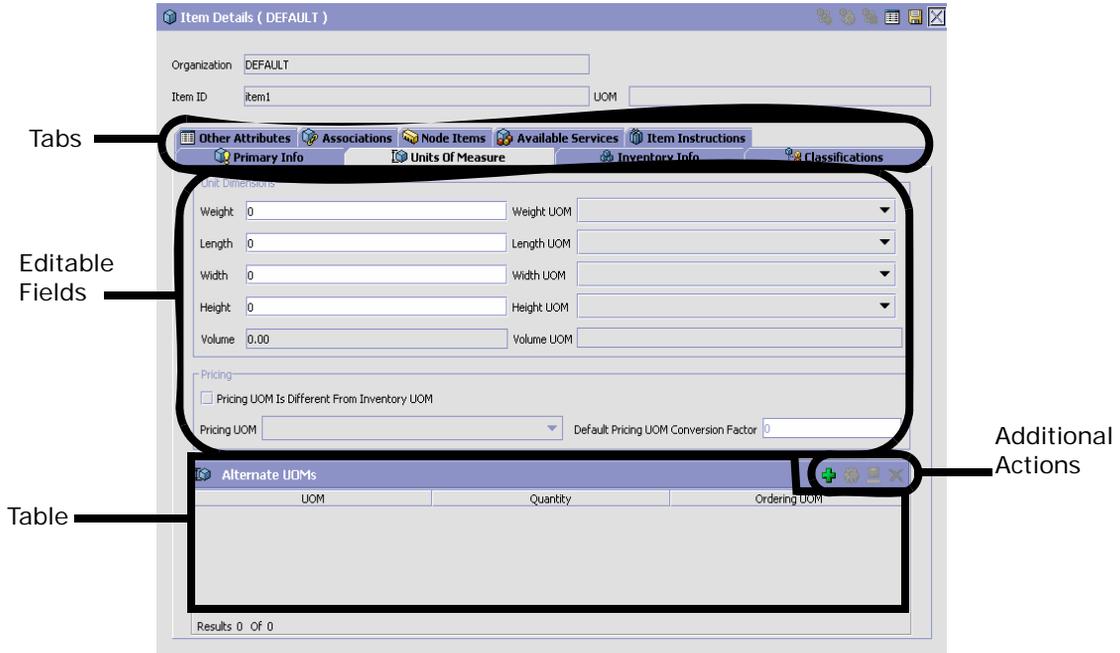
UOM Code	UOM Description
CM	Centimeter
FEET	Feet
IN	Inch
KM	Kilometer
METER	Meter
MILE	Mile

Below the table, the text "Results 6 Of 6" is displayed.

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.

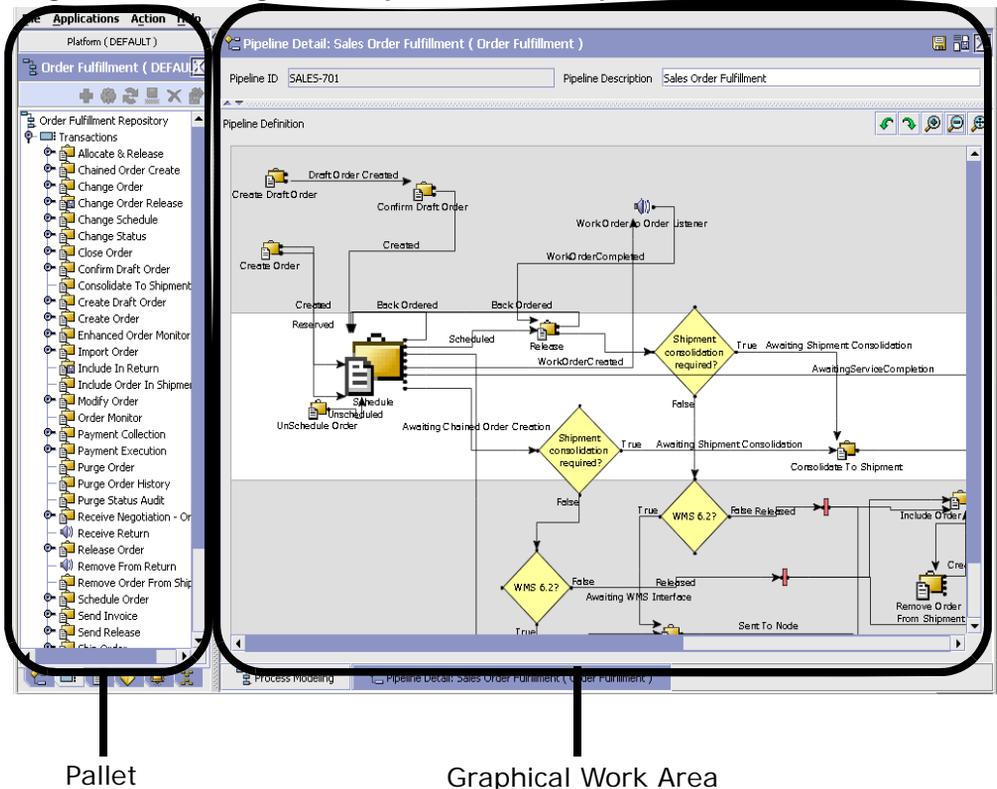
Figure 2–10 Details Window Example



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

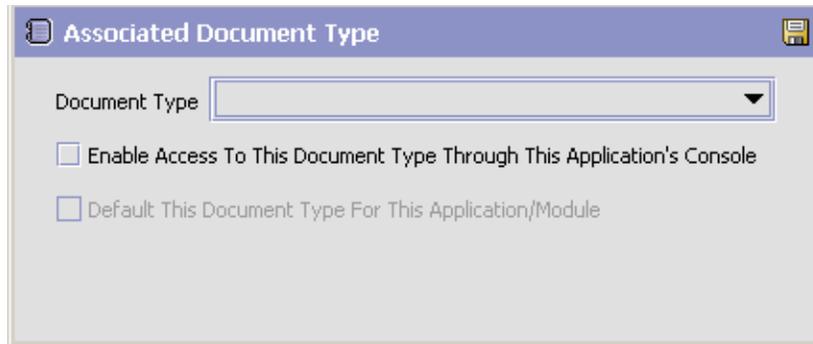
2.3.2.1 Adding a Document Type to an Application

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 .

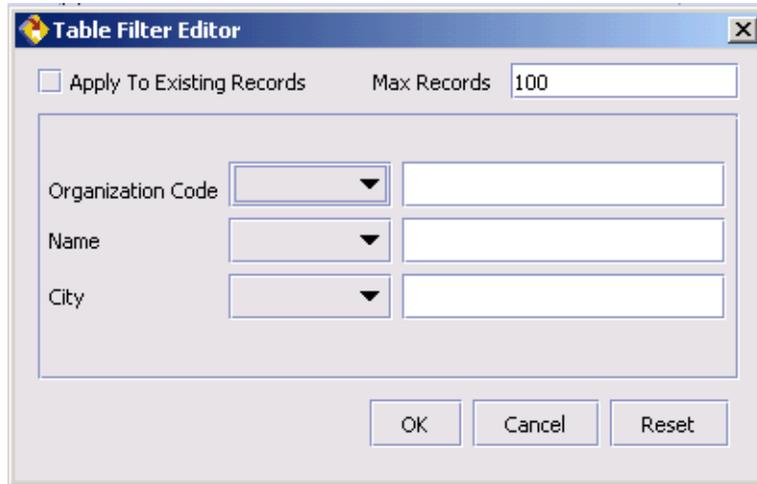
2.3.3 Viewing the User Logged into the Configurator

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 Participants

Trading partners using Yantra 7x to perform supply chain collaborative commerce are called *Participants*. Each Participant is considered an organization with a defined role.

You can use the Participant Modeling branch for:

- [Creating and Modifying an Organization](#)
- [Creating and Modifying an Organizational Hierarchy](#)
- [Creating Node Types](#)

3.1 Creating and Modifying an Organization

To create an organization:

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

Table 3–1 Create Organization Popup

Field	Description
Organization Code	Enter a unique code that identifies the organization.
Organization Name	Enter the name of the organization.
DUNS Number	Enter a unique nine-digit identification sequence, which provides unique identifiers of single business entities. Yantra 7x does not associate any logic with the DUNS number.
Account Number With Hub	If the organization is not the Hub, enter the account number that the organization has with the Hub.
Locale	Select the organization's geographic location.

Table 3–1 Create Organization Popup

Field	Description
Organization Is An Enterprise	Choose this to configure the organization as an Enterprise.
Primary Enterprise	<p>If the organization is not an Enterprise, choose this and select the applicable primary Enterprise.</p> <p>The primary enterprise will be defaulted on the entry point order console screens (for example, on search screens and create screens).</p> <p>On the organization details screen, when creating or modifying a node organization, the actions that appear on the primary info tab of the node attributes tab are the actions created for that enterprise. Whenever any enterprise level configuration is retrieved in the back-end business logic for a specific organization, the rules will always be retrieved for the primary enterprise of that organization.</p>

You can use the Organization Details window for:

- [Defining an Organization’s Primary Information](#)
- [Assigning the Organization’s Roles and Participant Associations](#)
- [Defining Communication Protocols](#)
- [Defining an Organization’s Payment Information](#)
- [Viewing an Organization’s Child Organizations](#)
- [Defining an Organization’s Calendars](#)
- [Viewing an Organization’s Departments](#)

3.1.1 Defining an Organization’s Primary Information

An organization’s primary information provides general information about the organization and the organization’s corporate and contact addresses.

To set up an organization’s primary information:

1. In the Organization Details window, choose the Primary Info tab.
2. Enter information in the applicable fields. Refer to [Table 3–2](#) for field value descriptions.

3. Choose .

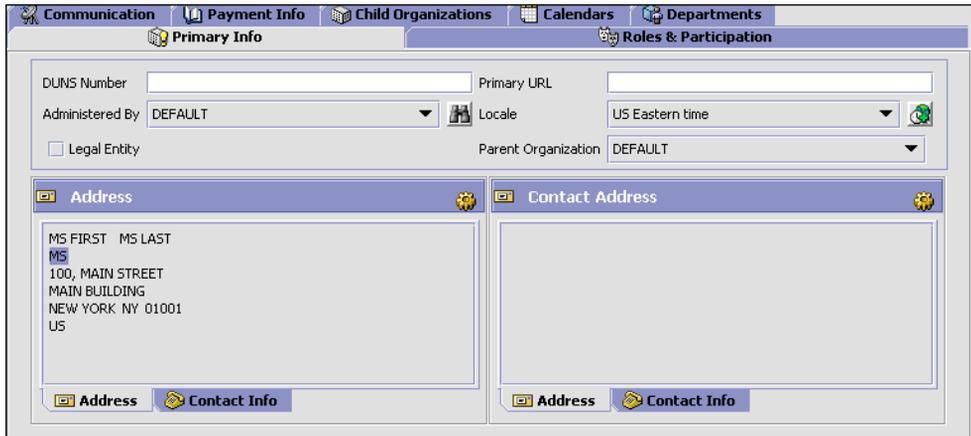


Table 3–2 Primary Info Tab

Field	Description
DUNS Number	Enter the unique nine-digit identification sequence, which provides unique identifiers of the business entity.
Primary URL	Enter the URL of the organization's Internet address, if applicable.
Administered By	Select the organization code of the organization that you want to administer this organization. Only the selected organization can make any changes to this organization in the Yantra 7x Configurator.
Locale	Select the organization's geographic location.

Table 3–2 Primary Info Tab

Field	Description
<p>Legal Entity</p>	<p>Select this field if this organization is its own legal entity.</p> <p>A legal entity is an organization unit identified by local governments as operating units and are typically instituted for each country a business operates in. The organizational unit is typically self-contained and is involved in recording all relevant transactions and generating all supporting documents for financial statements.</p> <p>For example, if you are a 3PL and have configured your 3PL as a Hub organization comprised of two Enterprises, one in the United States and one in India, both the US and Indian Enterprise organizations are their own legal entities.</p> <p>Important: If you are configuring a Seller organization, be aware that the Seller organization of one legal entity cannot fulfill orders by sourcing directly from another legal entity. In this case you would need to create a procurement purchase order with the organization.</p> <p>A procurement purchase order is a type chained order that is created when the final shipping point to the customer is a node within your organization and the shipping node does not have enough stock and needs to be replenished from an external organization's node.</p>
<p>Corporate Address</p>	<p>The address of the organization's corporate headquarters. This information is mandatory.</p> <p>Choose  to enter an address.</p> <p>Choose the Contact tab to view additional contact information.</p> <p>You can also specify Latitude and Longitude coordinates for this address.</p>

Table 3–2 Primary Info Tab

Field	Description
Primary Contact Address	<p>The address of the organization's main contact. This may be different than the Corporate Address. This information is mandatory.</p> <p>Choose  to enter an address.</p> <p>Choose the Contact tab to view additional contact information.</p> <p>You can also specify Latitude and Longitude coordinates for this address. If specified for a node, these coordinates will be used to plot the node on the Fulfillment Network Model. For more information on the Fulfillment Network Model, refer to the <i>Yantra 7x Distributed Order Management Configuration Guide</i>.</p>
Parent Organization	Choose in the organization hierarchy who is the parent organization.

3.1.2 Assigning the Organization's Roles and Participant Associations

For an organization to function as desired it must be given one or more roles. Each organization is assigned at least one role. A **role** is a well-defined set of activities that can be performed by an organization. Each organization performs at least one role. Yantra 7x supports the following organization roles:

- [Hub](#)
- [Enterprise](#)
- [Buyer](#)
- [Seller](#)
- [Carrier](#)
- [Node](#)

Hub

The *Hub* is the central organization around which all the other organizations are built is assigned the role of Hub. There is only one Hub. Typically the Hub is the entity that purchased Yantra 7x. The Hub

determines what kind of business model is used during configuration, for example, multi-divisional corporation, third-party logistics (3PL), or marketplace.

The Hub has the ability to configure the other organizations that interact with multiple Enterprises and assign their roles. The Hub also determines the document definitions available to all organizations and configures installation-level rules. The Hub can be assigned multiple roles, for example, Hub and Seller.

Enterprise

An Enterprise brokers business. Each organization in an organizational structure must be either an Enterprise or designate an Enterprise as its primary Enterprise.

Each Enterprise in a Hub can have many organizations that are assigned many roles.

Note: The Hub also acts as an Enterprise.

Whether or not an entity in a Hub is assigned an Enterprise role depends on the Hub's business model. For example, in the marketplace model, each market might be assigned an Enterprise role. This setup allows each market to be unique with their own product or service handling.

An Enterprise can define organizations that interact within their Enterprise. They can also set up document definitions to be available across all organizations and configure an Enterprise's carrier preferences.

An Enterprise in Yantra 7x controls the flow of order and logistics documents and is considered the owner of the various business documents throughout Yantra 7x.

An Enterprise defines most of the business rules and fulfillment processes for the orders. In many cases, such as a sales order, the Enterprise may be the Seller organization, and for purchase orders, the Enterprise may be the Buyer organization on the document. In other cases, if a higher level organizational unit wants to control and enforce business rules and document flow of all its subsidiaries, an Enterprise can represent this organizational unit and Seller/Buyer organization would be the subsidiary.

Buyer

An organization is assigned the **Buyer** role when it purchases product from the Enterprise or other organizations set up as Sellers.

A Buyer organization can participate with multiple Enterprises, but must assign a primary Enterprise with whom it participates. When it assigns its primary Enterprise, it takes on that Enterprise's inventory and catalog consolidation rules. When it interacts with multiple Enterprises it acts within the boundaries of the individual Enterprise's business rules.

Buyers can configure relationships with Seller organizations as well as Buyer services.

Seller

An organization is assigned the **Seller** role when it sells product to the Enterprise or other organizations set up as Buyers. Sellers can configure payment types, payment rules, and pricing for their organization.

When processing orders, a Seller organization can use the order, planned order, and purchase order process-type pipelines.

A seller organization can only see orders for which it is the buyer, seller or the enterprise.

Carrier

An organization is assigned the **Carrier** role when it provides delivery services between buyers, sellers, and customers. Special services, such as Next Day Air, can be offered dependant on the Carrier. United Parcel Service, Federal Express, and the United States Postal Service are all examples of carrier organizations.

Carriers can configure the services they provide such as truckload, less-than truckload, and parcel services.

Node

A **Node** represents a physical location (for example, a manufacturing plant, small stock room, or warehouse). A node can also play the role of Buyer or Seller.

A node organization is able to see orders for which its parent organization is the buyer, seller, enterprise, ship node, or receiving node.

Node roles are specified as follows:

- A child Node belongs to a parent organization. It cannot have any child organizations.
- A Buyer or Seller Node may belong to a parent organization, but it is not required to. It may have child organizations.

Note: If the organization you are creating participates in one or more Enterprises, you must identify each Enterprise as an associated participant.

To assign the organization's roles and participant associations:

1. In the Organization Details window, choose the Roles & Participation tab.



2. In the Roles box, select the roles that apply to this organization.
3. In the Enterprises box, choose + and select the Enterprises that this organization participates with from the Participating Enterprises pop-up window.
4. Select the primary Enterprise for the organization from the Enterprises box, if applicable.

Depending on the roles you chose for the organization, you may also have to use the Roles & Participation tab for:

- [Defining Enterprise Attributes](#)
- [Defining Seller Attributes](#)

- [Defining Buyer Attributes](#)
- [Defining Carrier Attributes](#)
- [Defining Node Attributes](#)
- [Defining an Organization's Advanced Inventory Attributes](#)
- [Defining an Organization's Advanced Catalog Attributes](#)
- [Defining an Organization's Advanced Capacity Attributes](#)

3.1.2.1 Defining Enterprise Attributes

If you chose Enterprise as a role for the organization, you can indicate the organization's that participate in the Enterprise as well as add and remove the participation with other organizations. You can also set up the Enterprise's carrier preferences.

You can use the Enterprise Attributes tab for:

- [Identifying Organizations that Participate in an Enterprise](#)
- [Modifying Organizations Associated with an Enterprise](#)
- [Removing an Organization from Participation in an Enterprise](#)
- [Defining an Enterprise's Print Format Preferences](#)
- [Setting Up an Enterprise's Cost Factor Preferences](#)

3.1.2.1.1 Defining an Enterprise's Primary Information

To define an Enterprise's primary information:

1. From the Roles & Participation tab in the Organization Details window, choose Enterprise Attributes.
2. In Enterprise Name, enter the name of the Enterprise you are configuring.
3. If want to inherit Enterprise level rules from another Enterprise, from Inherit Configuration From Enterprise, select the Enterprise whose configuration you want to inherit.

Important: If you do not specify an Enterprise to inherit from, you must configure all of the rules for the Enterprise you are creating.

- In UCC Prefix, enter the UCC code to be used when identifying the Enterprise on a shipment container marking (SCM).

3.1.2.1.2 Identifying Organizations that Participate in an Enterprise

Many organizations can participate in an Enterprise, such as a Carrier organization or a Buyer organization.

To identify organizations that participate in an Enterprise:

- From the Roles & Participation tab in the Organization Details window, choose Enterprise Attributes. The Participants list displays.

Organization Code	Name	City
AIRB	AIRB	
DEFAULT	Hub Organization	
FEDX	FEDX	
N1	N1	
UPSA	UPS Asia Pacific	
UPSC	UPS Canada	
UPSE	UPS Europe	
USPL	UPS Latin America	
USPN	United Parcel Service	
USPS	United States Postal Service	

Results 10 Of 10

- Choose . The Find Participants to Add Organization pop-up window appears.

Find Participants To Add Enterprise

Organization Code [dropdown] [text input] Roles [dropdown]

Organization Name [dropdown] [text input]

Enterprise [dropdown] Max Records 1,000

Search Results

Organization Code	Name	City
-------------------	------	------

Results 0

3. Enter the applicable search criteria and choose . A list of organizations displays.
4. Select the organization you want to participate in the Enterprise and choose .

3.1.2.1.3 Modifying Organizations Associated with an Enterprise

The Enterprise has the ability to modify an associated organization's details.

To modify an organization associated with an Enterprise:

1. From the Roles & Participation tab in the Organization Details window, choose Enterprise Attributes. The Participants list displays.
2. Select the applicable organization and choose . The Organization Details window for that organization displays.
3. Refer to topics throughout [Section 3.1, "Creating and Modifying an Organization"](#).
4. Choose .

3.1.2.1.4 Removing an Organization from Participation in an Enterprise

To remove an organization from participation in an Enterprise:

1. From the Roles & Participation tab in the Organization Details window, choose Enterprise Attributes. The Participants list displays.
2. Select the applicable Organization and choose .

3.1.2.1.5 Defining an Enterprise's Print Format Preferences

You can establish default print format preferences for documents and labels.

To set up an Enterprise's Print Format Preferences:

1. From the Roles & Participation tab in the Organization Details window, choose Enterprise Attributes Tab.
2. Select the Print Format Preferences Tab.
3. Enter information in the applicable fields. Refer to [Table 3–3](#) for field value descriptions.
4. Choose .

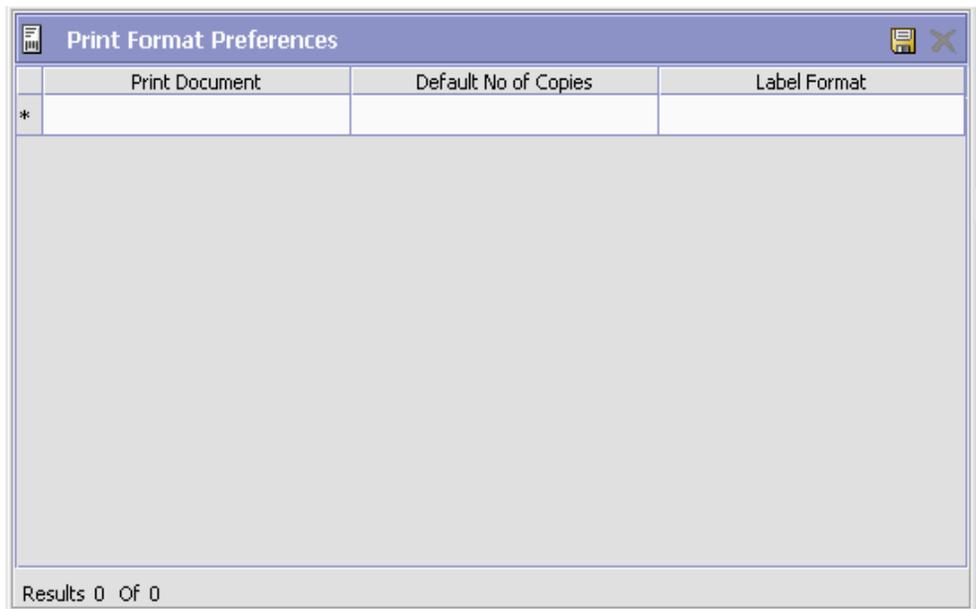


Table 3–3 Enterprise Attributes Print Format Preferences Tab

Field	Description
Print Document	From the drop down, select the print document for which you want to configure preferences.
Default No of Copies	Enter the number of copies that should be printed.
Label Format	Select which label format should be used with this document type.

3.1.2.1.6 Setting Up an Enterprise’s Cost Factor Preferences

You can identify the Inventory Costing Factors an Enterprise uses in its business model. If you define cost factors at the vendor level, you do not need to configure Enterprise Cost Factor Preferences.

To specify an Enterprise’s Cost Factor preferences:

1. From the Roles & Participation tab in the Organization Details window, choose Enterprise role.
2. Choose the Cost Factor Preferences tab. The Cost Factor Preferences list displays.



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

Table 3–4 Enterprise’s Cost Factor Preference List

Field	Description
Cost Factor Group to be Used for Standard Cost Calculations	Select the cost factor group you want to use for this enterprise’s standard cost calculations from the drop-down list.
Cost Factor Group to be Used for Landed Cost Calculations	Select the cost factor group you want to use for this enterprise’s landed cost calculations from the drop-down list.
Cost Factor Group to be Used for Physical Kit Cost Calculations	Select the cost factor group you want to use for this enterprise’s physical kit cost calculations from the drop-down list.

3.1.2.2 Defining Seller Attributes

If you choose Seller as a role for the organization, you indicate if the Seller requires payment processing and chained orders as well as set its default payment rule. On the Seller Attributes tab, if you choose Payment Processing Required, then you must also select a Default Payment Rule from the drop-down list.

To define a Seller organization’s attributes:

1. From the Roles & Participation tab in the Organization Details window, choose Seller Attributes.



2. Enter information in the applicable fields.

You can use the Seller Attributes tab, which holds information for the Seller Attributes, for:

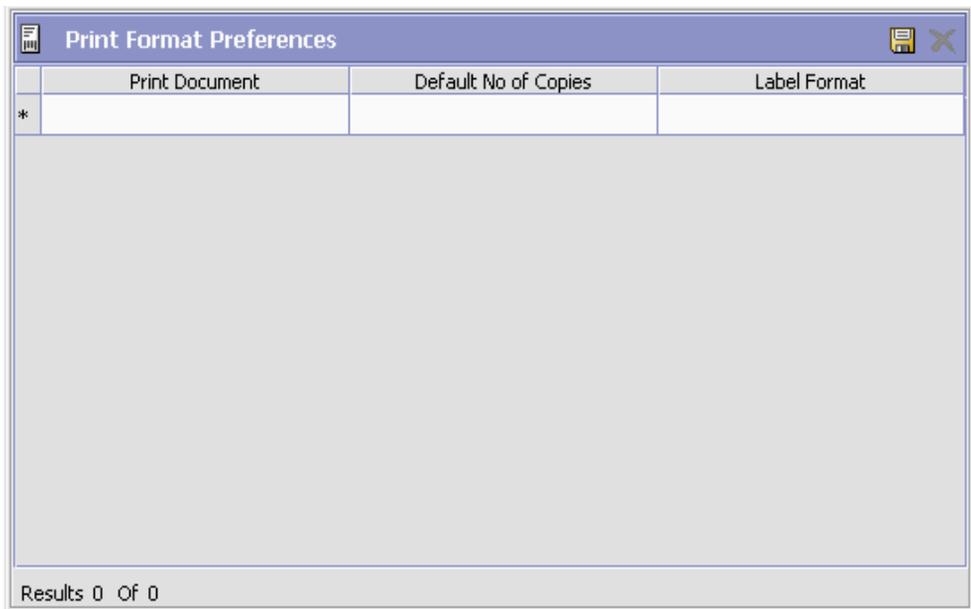
- [Defining a Seller’s Print Format Preferences](#)
- [Setting Up a Seller’s Cost Factor Preferences](#)

3.1.2.2.1 Defining a Seller’s Print Format Preferences

You can establish default print format preferences for documents and labels.

To set up a Seller's Print Format Preferences:

1. From the Roles & Participation tab in the Organization Details window, choose Seller Attributes Tab.
2. Select the Print Format Preferences Tab.
3. Enter information in the applicable fields. Refer to [Table 3–5](#) for field value descriptions.
4. Choose .



	Print Document	Default No of Copies	Label Format
*			

Results 0 Of 0

Table 3–5 Seller Attributes Print Format Preferences Tab

Field	Description
Print Document	From the drop down, select the print document for which you want to configure preferences.

Table 3–5 Seller Attributes Print Format Preferences Tab

Field	Description
Default No of Copies	Enter the number of copies that should be printed.
Label Format	Select which label format should be used with this document type.

3.1.2.2.2 Setting Up a Seller’s Cost Factor Preferences

You can identify the Inventory Costing Factors a Seller organization uses in its business model. If you define cost factors at the vendor level, you do not need to configure Enterprise Cost Factor Preferences.

To specify a Seller’s Cost Factor preferences:

1. From the Roles & Participation tab in the Organization Details window, choose the Seller role.
2. Choose the Cost Factor Preferences tab. The Cost Factor Preferences list displays.



3. Enter information in the applicable fields. Refer to [Table 3–6](#) for field value descriptions.
4. Choose .

Table 3–6 Seller’s Cost Factor Preference List

Field	Description
Cost Factor Group to be Used for Standard Cost Calculations	Select the cost factor group you want to use for this seller’s standard cost calculations from the drop-down list.
Cost Factor Group to be Used for Landed Cost Calculations	Select the cost factor group you want to use for this seller’s landed cost calculations from the drop-down list.

3.1.2.3 Defining Buyer Attributes

If you chose Buyer as a role for the organization, you can indicate the Buyer requirements for inbound compliance. Inbound compliance are the conditions the buyer has established for shipping and routing.

You can use the Inbound Compliance tab, which holds information for the Buyer attributes, for:

- [Defining Consolidation Parameters](#)
- [Defining Carrier Preferences Parameters](#)
- [Defining Routing Parameters](#)
- [Defining Packaging Parameters](#)
- [Defining Compliance Services Parameters](#)
- [Defining a Buyer's Print Format Preferences](#)
- [Defining a Buyer's Compliance Instructions](#)

3.1.2.3.1 Defining Consolidation Parameters

To set up a Buyer's Inbound Compliance Consolidation parameters:

1. From the Roles & Participation tab in the Organization Details window, choose the Inbound Compliance Tab.
2. Select the Consolidation Tab.
3. Enter information in the applicable fields. Refer to [Table 3–7](#) for field value descriptions.
4. Choose .

Enterprise Attributes Inbound Compliance Seller Attributes Advanced Attributes

Appointment Required

Do not mix in a shipment

Buyer Mark For Node Id Customer PO #

Department Code Gift Flag

Mark For Order #

Order Type

Transportation optimization

Economic shipping parameters maintained

Do not mix in a load

Addr Line 1 Addr Line 2

Addr Line 6 Name

Table 3–7 Inbound Compliance Consolidation Tab

Field	Description
Appointment Required	Check this box if the buyer requires an appointment to accept deliveries.
Do not mix in Shipment	If any of the following are selected, separate shipments must be created for items that have different values for these attributes. For example, if Department Code is selected, items that are for different departments can not be included in the same shipment.
Buyer MarkFor Node Identifier	Check this box if separate shipments must be created based on the markfor node identified by the buyer. For example, a markfor node is a store or a distribution center.
Customer PO #	Check this box if separate shipments must be created based on the Customer’s Purchase Order number.
Department Code	Check this box if separate shipments must be created based on the department for which the item is intended.
Gift Flag	Check this box if separate shipments must be created if the order line is a gift item.

Table 3–7 Inbound Compliance Consolidation Tab

Field	Description
Mark For	Check this box if separate shipments must be created based on the person for whom this shipment is marked for.
Order #	Check this box if separate shipments must be created based on the order number.
Order Type	Check this box if separate shipments must be created based on the buyer defined 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>
Expedite shipment by not more than ___ Days	<p>If shipments can be shipped earlier than their currently planned shipment date, Enter the number of days the shipment date can be moved forward.</p> <p>For example, if shipments can be shipped up to three days earlier than their current planned shipment date, enter '3'.</p>
Delay shipment by not more than ___ Days	Enter the number of days this shipment can be delayed before it should be shipped. 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.
Consolidate up to weight threshold of	Enter a weight.

Table 3–7 Inbound Compliance Consolidation Tab

Field	Description
Consolidate up to volume threshold of	Enter a volume.
Do not mix in a load	If any of the following are unselected, a shipment can be consolidated to the load, based on that attribute. For example, if Addr Line 1 and Name are unselected, shipments that have different first address line, but same address line 2 and 6 can be included in the same load.
Addr Line 1	Uncheck this box if you want to consolidate the shipment to a load based on the same first line of address.
Addr Line 2	Uncheck this box if you want to consolidate the shipment to a load based on the same second line of address.
Addr Line 6	Uncheck this box if you want to consolidate the shipment to a load based on the same sixth line of address.
Name	Uncheck this box if you want to consolidate the shipment to a load based on the first, middle and last name.

3.1.2.3.2 Defining Carrier Preferences Parameters

To set up a Buyer’s Inbound Compliance Carrier Preferences parameters:

1. From the Roles & Participation tab in the Organization Details window, choose the Inbound Compliance Tab.
2. Select the Parcel Carrier Preferences Tab.

Defining Carrier Preferences is described in [Section 3.1.2.5.3, "Defining a Node's Parcel Carrier Preferences"](#) on page 90. Use those procedures to define and maintain the Carrier preferences for the Buyer.

3.1.2.3.3 Defining Routing Parameters

The Routing Parameters for a Buyer consist of routing guides. *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.

The Routing Parameters tab can be used for:

- [Setting Routing Preferences](#)
- [Creating a Routing Guide](#)
- [Modifying a Routing Guide](#)
- [Deleting a Routing Guide](#)

Setting Routing Preferences

To set the over all routing preferences of the buyer

1. From the Roles & Participation Tab in the Organization Details window, choose the Inbound Compliance Tab.
2. Select the Routing Tab.
3. Set the Routing Guide value, as described in [Table 3–8](#).

Table 3–8 Routing Guide Preferences

Fields	Description
Region Schema For Routing	Select the applicable region schema for routing. Select  . The Region Schema Details window appears. For more information about modifying the region schema details, see Section 13.3, "Defining Region Schemas" on page 357.
Routing Guide	
Not Maintained	Select this to use manual routing. Shipments are managed in the shipment console, and any routing guides are not consulted.

Table 3–8 Routing Guide Preferences

Fields	Description
Maintained in Yantra	<p>Select this to use the Routing Guides maintained in Yantra 7x to determine how shipments should be routed.</p> <p>In addition to the routing guide maintained here by the enterprise, there may be a routing guide for the buyer organization.</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>

Creating a Routing Guide

To set up a Buyer’s Inbound Compliance Routing parameters:

1. From the Roles & Participation Tab in the Organization Details window, choose the Inbound Compliance Tab.
2. Select the Routing Tab.
3. Select  on the Routing Guides list window. The Routing Guide Details window appears in the work area.
4. Enter information in the applicable fields. Refer to [Table 3–9](#) for field value descriptions.
5. Choose .

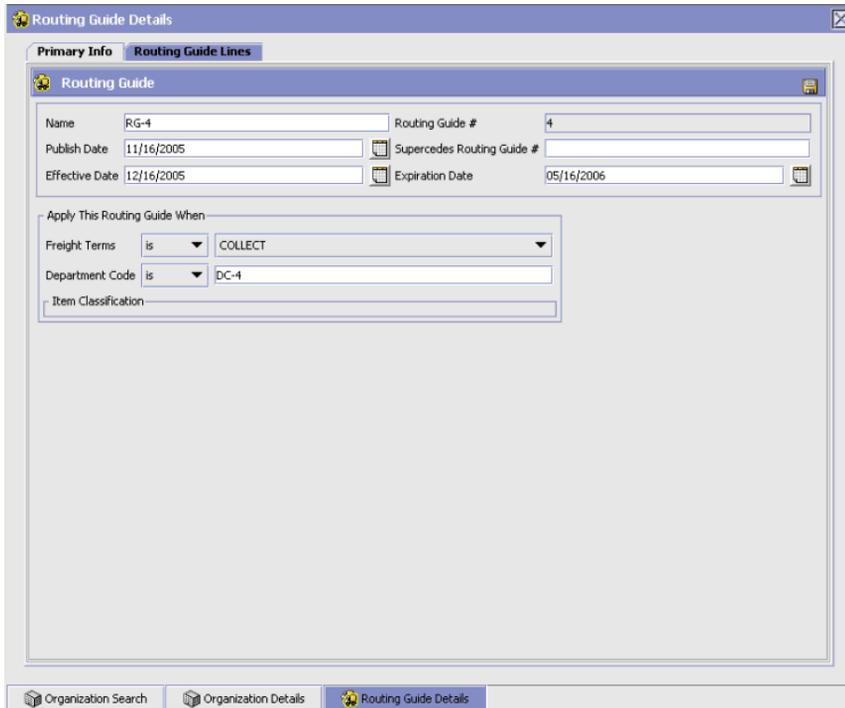


Table 3–9 Routing Guide Details Window

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.
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.
Supercedes 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 supercedes routing guide "1234.
Expiration Date	The end date for applying the routing information in this routing guide.

Table 3–9 Routing Guide Details Window

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

The Routing Guide Details Window can be used for:

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

Modifying a Routing Guide

To set up a Buyer’s Inbound Compliance Routing parameters:

1. From the Roles & Participation Tab in the Organization Details window, choose the Inbound Compliance Tab.
2. Select the Routing Tab.
3. Select a routing guide in the Routing Guide list window, and select .

4. The Routing Guide Details window appears in the work area.
5. Enter information in the applicable fields. Refer to [Table 3–9](#) for field value descriptions.
6. Choose .

Deleting a Routing Guide

To delete a routing guide:

To set up a Buyer's Inbound Compliance Routing parameters:

1. From the Roles & Participation Tab in the Organization Details window, choose the Inbound Compliance Tab.
2. Select the Routing Tab.
3. Select the applicable Routing Guide and choose .

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.

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).

To create a routing guideline:

1. From the Routing Guide Details window, select the Routing Guidelines Tab. To have access to the Routing Guidelines Tab, save the information you have entered on the Primary Info Tab.
2. A Routing Guide Line search window appears.
3. Select . A Routing Guide Line Details screen appears in the work area.
4. Enter information in the applicable fields. Refer to [Table 3–10](#) for field value descriptions.

5. Choose .

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:

Priority	Carrier/Service	Break Bulk Node	Contact Specified
Results 0 Of 0			

With overrides:

Override Freight Terms

Override Ship To

Table 3–10 Routing Guide Line Details Window

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

Table 3–10 Routing Guide Line Details Window

Consolidator	Select the consolidator name.
Store#	Select the store number.
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 containers.
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.
Then ship via:	
For details about defining carrier services, see Section 3.1.2.3.4, "Defining Carrier Services" on page 60.	
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. This 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 node that is close to the buyer.
Contact Specified	Indicates whether the contact details for the shipment is specified.

Table 3–10 Routing Guide Line Details Window

With overrides:	
Override Freight Terms	Select to override the shipment's Freight Term.
Override Ship To	To override the Ship To value, select this field, and then select one of the following. This is only used when performing routing again due to a revised ship to address.
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.

3.1.2.3.4 Defining Carrier Services

When routing occurs, the shipment is matched against the routing guidelines. Based on the criteria specified, you select a carrier service to use.

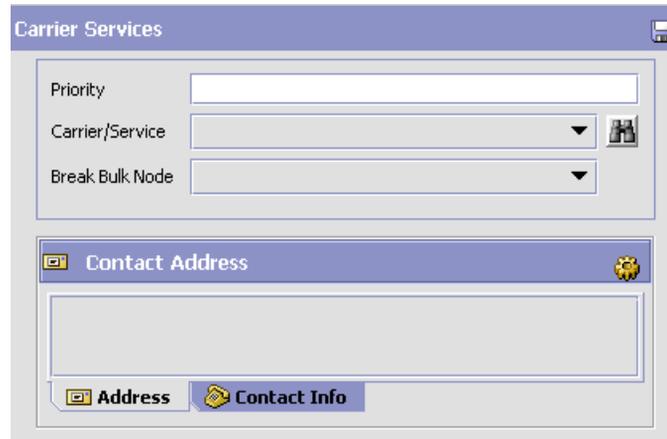
You can use the Carrier Services panel for:

- [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.



2. Enter information in the applicable fields. Refer to [Table 3–11](#) for field value descriptions.
3. Choose .

Table 3–11 *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. This 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 node that is close to the buyer.
Contact Address	This is used to specify the address information for the carrier service's contact person. 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 from the list in the Carrier Services list window, and select . The Carrier Services window appears.
2. Enter the new information in the applicable fields. Refer to [Table 3–11](#) 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 .
2. Choose .

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 3–10](#) for field value descriptions.
4. Choose .

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 .

3.1.2.3.5 Defining Packaging Parameters

To set up a Buyer's Inbound Compliance Packaging parameters:

1. From the Roles & Participation tab in the Organization Details window, choose the Inbound Compliance Tab.
2. Select the Packaging Tab.
3. Enter information in the applicable fields. Refer to [Table 3–12](#) for field value descriptions.
4. Choose .

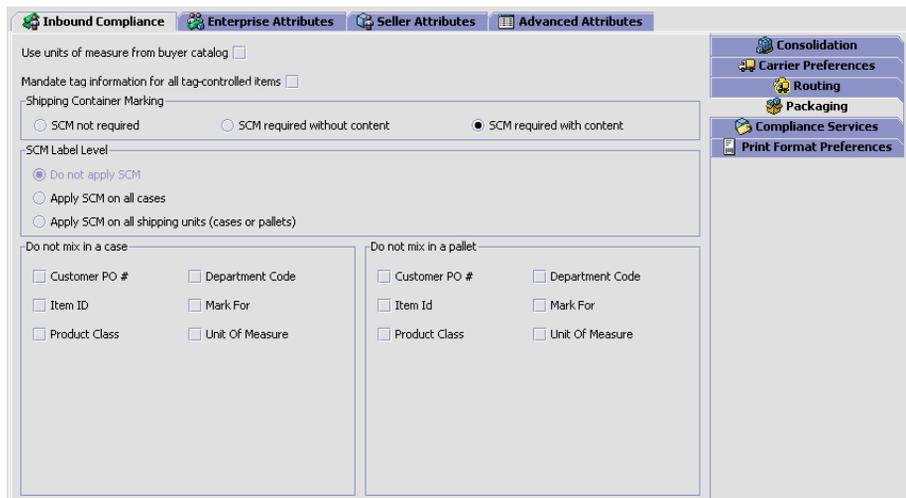


Table 3–12 Inbound Compliance Packaging Tab

Field	Description
Use units of measure from buyer catalog	The buyer may specify units of measure, such as the number of packages contained in a case, or the number of cases in a pallet in the buyer catalog. Select this field to ensure that shipments match these requirements. If this field is not selected, then the enterprise's units of measure are used.
Mandate tag information for all tag-controlled items	The buyer may mandate tag information for all tag-controlled items. Select this field to ensure that the tag information is provided whenever you ship tag-controlled items to a buyer.

Table 3–12 Inbound Compliance Packaging Tab

Field	Description
Shipping Container Marking	
SCM not required	Select this field if a shipping container marking is not required on a container.
SCM required without content	Select this field if a shipping container marking is required on a container but does not have to contain information.
SCM required with content	Select this field if a shipping container marking is required on a container with all applicable information filled out.
SCM Label Level	
Do not apply SCM	If you selected 'SCM not required', this field is automatically chosen.
Apply SCM on all cases	If you specified that shipment container markings are required, choose this field if you want to apply them on cases only.
Apply SCM on all shipping units (cases or pallets)	If you specified that shipment container markings are required, choose this field if you want to apply them on cases and pallets.
Do not mix in a case	
Customer PO #	Select this field if you do not want SKUs for shipments with different customer purchase order numbers to be packed into the same case.
Department Code	Select this field if you do not want SKUs for shipments with different department codes to be packed into the same case.
Item ID	Select this field if you do not want SKUs with different item IDs to be packed into the same case.
Mark For	Select this field if you do not want SKUs for shipments with different mark for addresses to be packed into the same case.
Unit Of Measure	Select this field if you do not want SKUs with different units of measure to be packed into the same case.
Product Class	Select this field if you do not want SKUs with different product classes to be packed into the same case.
Do not mix in pallet	

Table 3–12 Inbound Compliance Packaging Tab

Field	Description
Customer PO #	Select this field if you do not want SKUs for shipments with different customer purchase order numbers to be packed onto the same pallet.
Department Code	Select this field if you do not want SKUs for shipments with different department codes to be packed onto the same pallet.
Item ID	Select this field if you do not want SKUs with different item IDs to be packed onto the same pallet.
Mark For	Select this field if you do not want SKUs for shipments with different mark for addresses to be packed onto the same pallet.
Unit Of Measure	Select this field if you do not want SKUs with different units of measure to be packed onto the same pallet.
Product Class	Select this field if you do not want SKUs with different product classes to be packed onto the same pallet.

3.1.2.3.6 Defining Compliance Services Parameters

Compliance service parameters describe additional services that a buyer wants performed on certain items. A few examples are customization of the items, such as placing a security tag on the item, adding promotional materials in the item’s packaging, or monogramming the item with a corporate logo.

The Compliance Services Tab can be used for:

- [Creating Compliance Services](#)
- [Modifying Compliance Services](#)
- [Deleting Compliance Services](#)

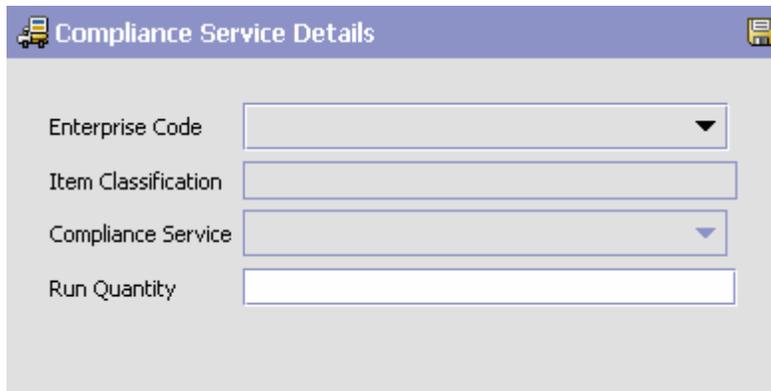
To set up a Buyer’s Inbound Compliance Services parameters:

1. From the Roles & Participation tab in the Organization Details window, choose Inbound Compliance Tab.
2. Select the Compliance Services Tab.
3. To enable the use of Compliance services, select the Requires VAS Compliance box, and specify one or more compliance services that the Buyer wants to be performed on items.

Creating Compliance Services

To create a Compliance Service:

1. From the Compliance Services Tab, select .
2. In the Compliance Service Details popup that appears, enter information in the applicable fields. Refer to [Table 3–13](#) for field value descriptions.
3. Choose .



The screenshot shows a 'Compliance Service Details' popup window. The title bar is blue and contains a printer icon on the left and a save icon on the right. The main area is light gray and contains four input fields:

- Enterprise Code:** A dropdown menu with a downward arrow.
- Item Classification:** A text input field.
- Compliance Service:** A dropdown menu with a downward arrow.
- Run Quantity:** A text input field.

Table 3–13 Compliance Services Details

Field	Description
Enterprise Code	<p>Indicates the Enterprise for which these criteria should be applied. A buyer may set up different compliance requirements interacting with different enterprises.</p> <p>The Enterprise selected determines the available service items and classifications based on their catalog organization.</p>
Item Classification	<p>The classification code for a group of items. For example, if a security tag is applied to jewelry worth more than \$200, you may establish an Item Classification “Expensive Jewelry”, and then classify items as being members of the “Expensive Jewelry” classification.</p> <p>When an item of expensive jewelry is ordered, the compliance service is executed. The compliance service defines what compliance services should be performed on the jewelry, including adding the security tag.</p>

Table 3–13 Compliance Services Details

Field	Description
Compliance Service	<p>Identifies the compliance service. The compliance service describes a series of activities that are performed to meet the buyer's requirements. These could include adding custom logos, inserting advertisements, using seasonal promotion boxes, or many other activities which customize the item for the buyer.</p>
Run Quantity	<p>The number of items to be made when a compliance service is run. By grouping the production of an item that has compliance services applied to it, inventory can be created in anticipation of the buyer's need.</p> <p>The run quantity is a number that indicates how many items to batch together: the actual request for product and available inventory determine how many items should have the compliance service applied.</p> <p>For example, if the run quantity is 10, the buyer requests 8 of the item, and there's only 1 on hand, 10 items will have the compliance services applied. The result is the buyer receives 8 items, 1 from current inventory, 7 that is newly created, and there will be 3 newly created items now available in inventory.</p> <p>If the buyer requires more than the run quantity will produce, the run quantity is used to create several runs. For example, if the buyer were to request 22 items, and only 1 item is in stock, doing a run of 10 would not satisfy the request. Doing two runs of 10 each would still not satisfy the request, but doing 3 runs would satisfy the request. Therefore, a single run of 30 is done.</p> <p>The run quantity should be set based on the anticipated buyer requirements. For example, for bulky items such as refrigerators or washing machines, the number might be low. For items that the buyer purchases in large quantities, such as T-shirts embroidered with a sports team logo, the number might be higher.</p> <p>Note: When the run quantity is not equal to an actual work order's ordered quantity, an additional inventory check is performed during scheduling.</p>

Modifying Compliance Services

To modify a Compliance Service:

1. From the Inbound Compliance window, select the Compliance Services Tab.
2. Select a Compliance Service from the Compliance Services list, and select . The Compliance Service Details popup window appears.
3. Enter information in the applicable fields. Refer to [Table 3–13](#) for field value descriptions.
4. Choose .

Deleting Compliance Services

1. From the Inbound compliance window, select the Compliance Services Tab.
2. Select a Compliance Service from the Compliance Services list, and select .

3.1.2.3.7 Defining a Buyer's Print Format Preferences

You can establish default print format preferences for documents and labels.

To set up a Buyer's Inbound Compliance Print Format Preferences:

1. From the Roles & Participation tab in the Organization Details window, choose the Inbound Compliance Tab.
2. Select the Print Format Preferences Tab.
3. Enter information in the applicable fields. Refer to [Table 3–14](#) for field value descriptions.
4. Choose .

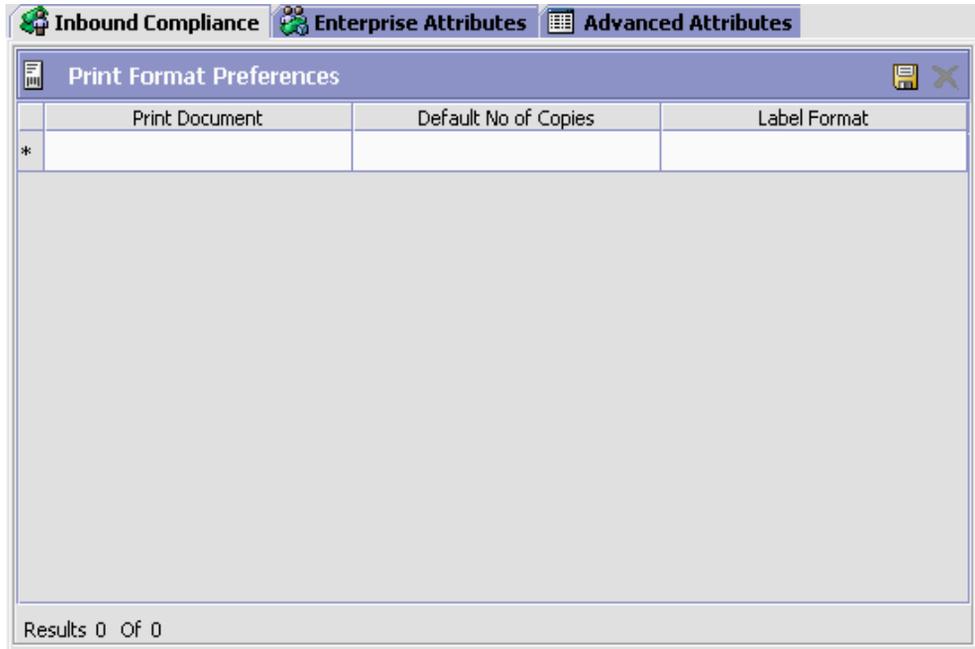


Table 3–14 Inbound Compliance Print Format Preferences Tab

Field	Description
Print Document	From the drop down, select the print document for which you want to configure preferences.
Default No of Copies	Enter the number of copies that should be printed.
Label Format	Select which label format should be used with this document type.

3.1.2.3.8 Defining a Buyer’s Compliance Instructions

You can define any special instructions that are appropriate to a buyer. You can print these instructions on a packing slip or a specific instruction sheets. For example, you can specify that the height of a carton should not be more than 15 inches.

You can use the Compliance Instructions tab for:

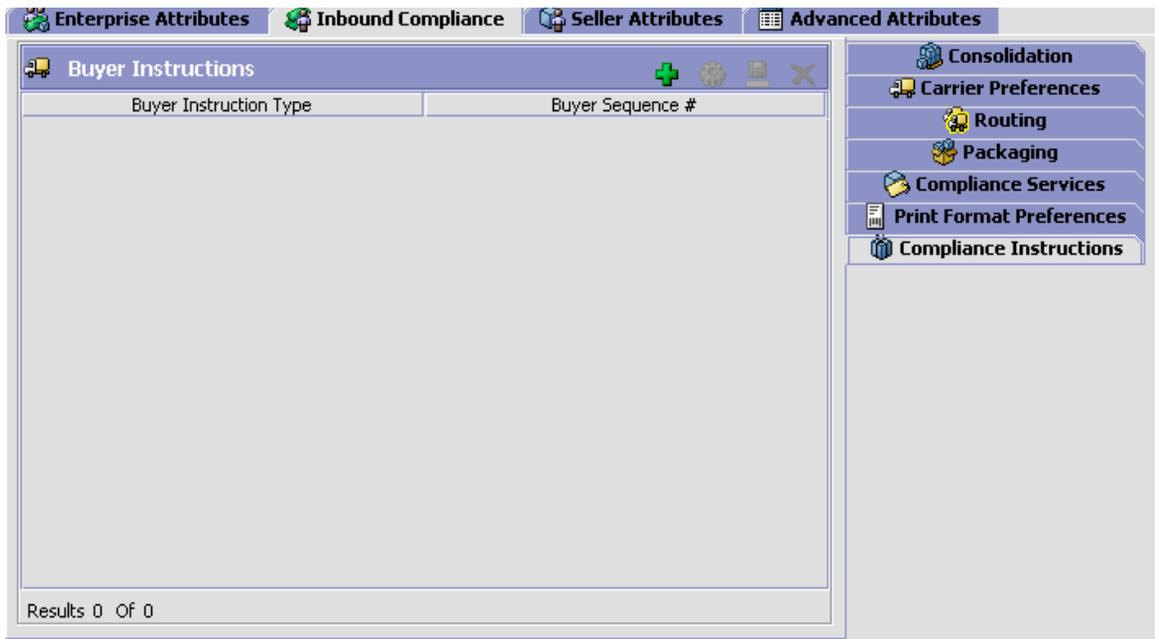
- [Creating a Buyer Instruction](#)

- [Modifying a Buyer Instruction](#)
- [Deleting a Buyer Instruction](#)

Creating a Buyer Instruction

To create a Buyer Instruction:

1. In the Organization Details window, from the Roles & Participation tab, choose the Inbound Compliance Tab.
2. Select the Compliance Instructions Tab.



3. From the Buyer Instructions table, choose . The Buyer Instruction Details pop-up window appears.
4. Enter information in the applicable fields. Refer to [Table 3–15](#) for field value descriptions.
5. Choose .

Table 3–15 Inbound Compliance Buyer Instructions Tab

Field	Description
Buyer Instruction Type	Select the buyer instruction type.
Sequence No	If there are multiple instructions for the buyer, enter the sequence number of the instruction you are adding.

Modifying a Buyer Instruction

To modify a Buyer Instruction:

1. In the Inbound Compliance tab, choose the Compliance Instructions tab.
2. From the Buyer Instructions table, select the applicable item instruction choose . The Buyer Instruction Details pop-up window appears.
3. Enter information in the applicable fields. Refer to [Table 3–15](#) for field value descriptions.
4. Choose .

Deleting a Buyer Instruction

To delete a Buyer Instruction:

1. In the Inbound Compliance Instructions tab, choose the Compliance Instructions tab.

- From the Buyer Instructions table, select the applicable buyer instruction and choose .

3.1.2.4 Defining Carrier Attributes

If you chose Carrier as a role for the organization, you can specify the types of services a Carrier provides for truckload, less-than truckload, and parcel shipments. A carrier service is the service a Carrier provides for the delivery of an order, such as, Second Day Air.

You can use the Carrier Attributes tab for:

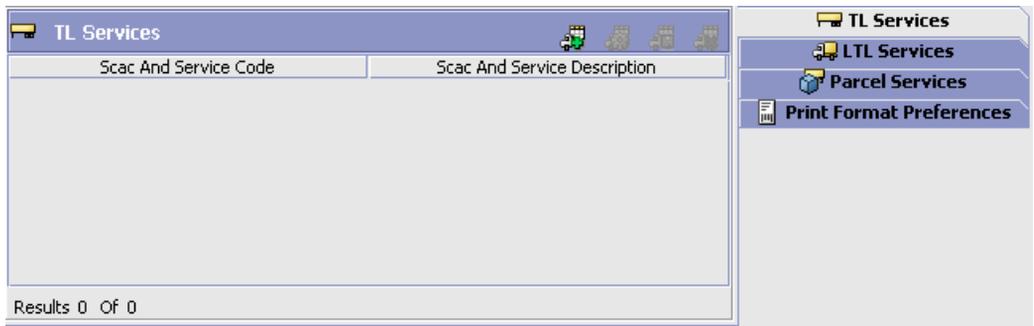
- [Defining Carrier Services for Truckload Shipments](#)
- [Defining Carrier Services for Less-Than Truckload Shipments](#)
- [Defining Carrier Services for Parcel Shipments](#)

3.1.2.4.1 Defining Carrier Services for Truckload Shipments

A Truckload is the largest of the shipping modes. A truckload is normally considered a shipment of over 10,000 pounds. You can add, modify, and delete carrier services for the Truckload ship mode.

To create carrier services for truckload shipments:

- From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
- Choose the TL Services tab, the Truckload Services list displays.



- Choose . The Truckload Service Details pop-up window appears.
- Enter information in the applicable fields. Refer to [Table 3–16](#) for field value descriptions.

5. Choose .

Table 3–16 Truckload Service Details Pop-Up Window

Field	Description
Service	Choose the carrier service code.
Scac and Service	Enter the name of the SCAC and service. Note: This is populated when the carrier service code is chosen, and it is editable.
Scac and Service Description	Enter a brief description of the carrier SCAC and service. Note: This is populated when the carrier service code is chosen, and it is editable.
Electronic Code	Enter the code used by the carrier organization to identify the service. For example, UPS Next Day Air has the electronic code '01'.
Host Code	This field is not used in this version of Yantra 7x.
Extended Ship Mode	If the carrier service is not TL, LTL, or parcel, select the correct shipment mode.

Table 3–16 Truckload Service Details Pop-Up Window

Field	Description
Fixed Transit Days	<p>Enter the maximum number of days 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>This number is used for order line scheduling. This value is only used if the Use Advanced Transit Time Calculation flag on the Other Rules tab under Distributed Order Management > Cross Application > Logistics > 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 the service travels each transit day. Choose the relevant UOM for the distance.</p> <p>This number is used for order line scheduling. This value is only used if the Use Advanced Transit Time Calculation flag on the Other Rules tab under Distributed Order Management > Cross Application > Logistics > Logistics Attributes is selected. For more information about this field, see the <i>Yantra 7x Distributed Order Management Configuration Guide</i>.</p>
Allow Shipping Of Hazardous Materials	Check this box to indicate that this carrier allows shipping of hazardous materials.
Ship By Air	Check this box to indicate that this carrier can ship by air.
Has Freezer	Check this box to indicate that this carrier can ship items that require freezer storage.
Package Level Integration	Select this option if this carrier service caters to package level integration.
Shipment Level Integration	Select this option if this carrier service caters to shipment level integration.

To modify carrier services for truckload shipments:

1. From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
2. Choose the TL Services tab, the Truckload Services list displays.
3. Select the applicable carrier service and choose . The Truckload Service Details pop-up window appears.

4. Modify information in the applicable fields. Refer to [Table 3–16](#) for field value descriptions.
5. Choose .

To delete a carrier service for truckload shipments:

1. From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
2. Choose the TL Services tab, the Truckload Services list displays.
3. Select the applicable carrier service and choose .

3.1.2.4.2 Defining Carrier Services for Less-Than Truckload Shipments

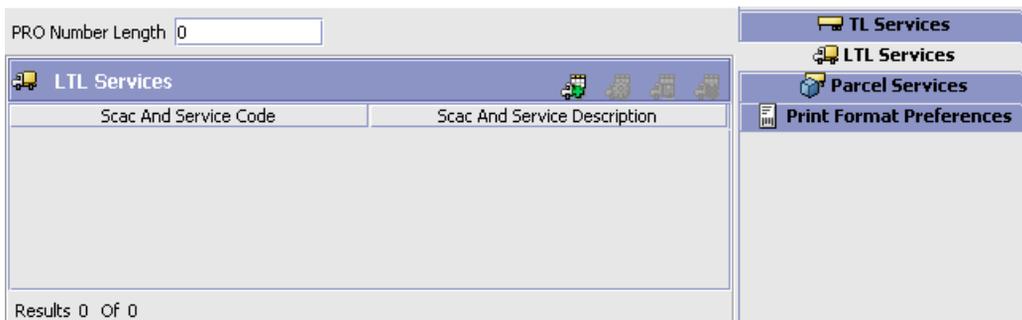
A shipment is normally referred to as a less-than truckload shipment when it weighs between 150 and 10,000 pounds. You can add, modify, and delete carrier services for the less-than truckload shipping mode.

To create carrier services for less-than truckload shipments:

1. From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
2. Enter the PRO Number Length. The PRO Number Length refers to the number of digits in the PRO Number.

For more information about PRO Numbers and their generation, see [Section 3.1.2.5.5, "Defining a Node's LTL Carrier Preferences"](#) on page 96.

3. Choose the LTL Services tab, the Less-Than Truckload Services list displays.



PRO Number Length

Scac And Service Code	Scac And Service Description
-----------------------	------------------------------

Results 0 Of 0

- TL Services
- LTL Services
- Parcel Services
- Print Format Preferences

4. Choose . The Less-Than Truckload Service Details pop-up window appears.
5. Enter information in the applicable fields. Refer to [Table 3–17](#) for field value descriptions.
6. Choose .

Table 3–17 *Less-Than Truckload Service Details Pop-Up Window*

Field	Description
Service	Choose the carrier service code.
Scac and Service	Enter the name of the SCAC and service. Note: This is populated when the carrier service code is chosen, and it is editable.
Scac and Service Description	Enter a brief description of the carrier SCAC and service. Note: This is populated when the carrier service code is chosen, and it is editable.
Electronic Code	Enter the code used by the carrier organization to identify the service. For example, UPS Next Day Air has the electronic code '01'.

Table 3–17 Less-Than Truckload Service Details Pop-Up Window

Field	Description
Host Code	This field is not used in this version of Yantra 7x.
Extended Ship Mode	If the carrier service is not TL, LTL, or parcel, select the correct shipment mode.
Fixed Transit Days	<p>Enter the maximum number of days 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>This number is used for order line scheduling. This value is only used if the Use Advanced Transit Time Calculation flag on the Other Rules tab under Distributed Order Management > Cross Application > Logistics > 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. Choose the relevant UOM for the distance.</p> <p>This number is used for order line scheduling. This value is only used if the Use Advanced Transit Time Calculation flag on the Other Rules tab under Distributed Order Management > Cross Application > Logistics > Logistics Attributes is selected. For more information about this field, see the <i>Yantra 7x Distributed Order Management Configuration Guide</i>.</p>
Allow Shipping Of Hazardous Materials	Check this box to indicate that this carrier allows shipping of hazardous materials.
Ship By Air	Check this box to indicated that this carrier can ship by air.
Has Freezer	Check this box to indicate that this carrier can ship items that require freezer storage.
Package Level Integration	Select this option if this carrier service caters to package level integration.
Shipment Level Integration	Select this option if this carrier service caters to shipment level integration.

To modify carrier services for less-than truckload shipments:

1. From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
2. Choose the LTL Services tab, the Less-Than Truckload Services list displays.
3. Edit the PRO Number Length. The PRO Number Length refers to the number of digits in the PRO Number.

For more information about PRO Numbers and their generation, see [Section 3.1.2.5.5, "Defining a Node's LTL Carrier Preferences"](#) on page 96.

4. Select the applicable carrier service and choose . The Less-Than Truckload Service Details pop-up window appears.
5. Modify information in the applicable fields. Refer to [Table 3–17](#) for field value descriptions.
6. Choose .

To delete a carrier service for less-than truckload shipments:

1. From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
2. Choose the LTL Services tab, the Less-Than Truckload Services list displays.
3. Select the applicable carrier service and choose .

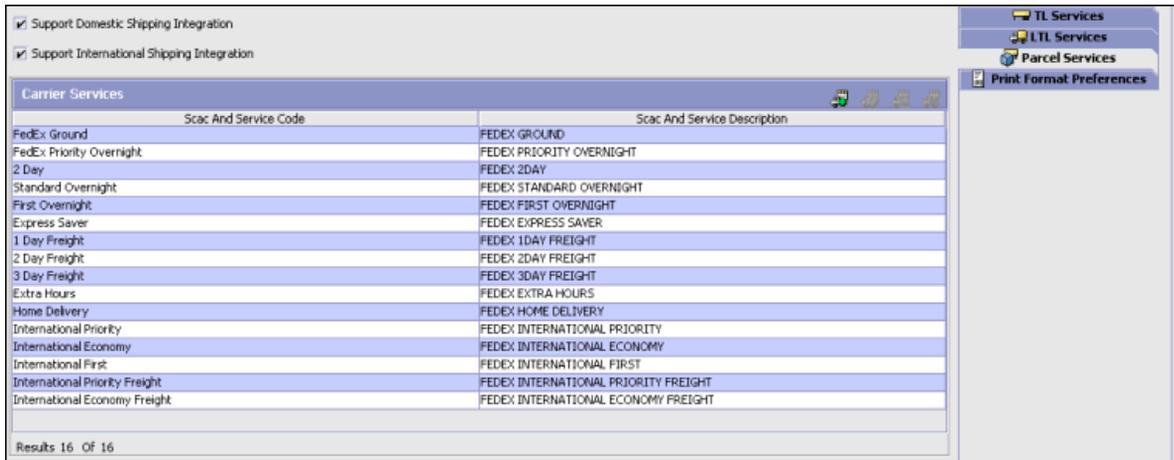
3.1.2.4.3 Defining Carrier Services for Parcel Shipments

A parcel is the smallest of the shipping modes. A shipment is normally referred to as a parcel when it weighs under 150 pounds. You can add, modify, and delete carrier services for the parcel shipping mode.

To create carrier services for parcel shipments:

1. From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
2. Choose the Parcel Services tab, the Parcel Services list displays.
3. Check the Support Domestic Shipping Integration box, if you want to integrate through the carrier's external systems for domestic shipments.

4. Check the Support International Shipping Integration box, if you want to integrate through the carrier's external systems for international shipments.



5. Choose . The Parcel Service Details pop-up window appears.
6. Enter information in the applicable fields. Refer to [Table 3–18](#) for field value descriptions.
7. Choose .

Table 3–18 Parcel Service Detail Pop-Up Window

Field	Description
Service	Choose the carrier service code.
Scac and Service Code	Enter the name of the SCAC and service. Note: This is populated when the carrier service code is chosen, and is editable.
SCAC and Service Description	Enter a brief description of the carrier SCAC and service. Note: This is populated when the carrier service code is chosen, and is editable.
Electronic Code	Enter the code used by the carrier organization to identify the service. For example, UPS Next Day Air has the electronic code '01'.
Host Code	This field is not used in this version.
Extended Ship Mode	If the carrier service is not TL, LTL, or parcel, select the correct shipment mode.

Table 3–18 Parcel Service Detail Pop-Up Window

Field	Description
Fixed Transit Days	<p>Enter the maximum number of days 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>This number is used for order line scheduling. This value is only used if the Use Advanced Transit Time Calculation flag on the Other Rules tab under Distributed Order Management > Cross Application > Logistics > 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 the service travels each transit day. Choose the relevant UOM for the distance.</p> <p>This number is used for order line scheduling. This value is only used if the Use Advanced Transit Time Calculation flag on the Other Rules tab under Distributed Order Management > Cross Application > Logistics > Logistics Attributes is selected. For more information about this field, see the <i>Yantra 7x Distributed Order Management Configuration Guide</i>.</p>
Allow Shipping Of Hazardous Materials	Check this box to indicate that this carrier allows shipping of hazardous materials.
Ship By Air	Check this box to indicated that this carrier can ship by air.
Has Freezer	Check this box to indicate that this carrier can ship items that require freezer storage.
Package Level Integration	Select this option if this carrier service caters to package level integration.
Shipment Level Integration	Select this option if this carrier service caters to shipment level integration.

To modify carrier services for parcel shipments:

1. From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
2. Choose the Parcel Services tab, the Parcel Services list displays.
3. Select the applicable carrier service and choose . The Parcel Service Details pop-up window appears.

4. Modify information in the applicable fields. Refer to [Table 3–18](#) for field value descriptions.
5. Choose .

To delete a carrier service for parcel shipments:

1. From the Roles & Participation tab in the Organization Details window, choose Carrier Attributes.
2. Choose the Parcel Services tab, the Parcel Services list displays.
3. Select the applicable carrier service and choose .

3.1.2.5 Defining Node Attributes

If you chose Node as a role for the organization, you can specify its primary information, sourcing, scheduling, carrier preferences, and calendars.

You can use the Node Attributes tab for:

- [Defining a Node's Primary Information](#)
- [Defining a Node's Sourcing and Scheduling](#)
- [Defining a Node's Parcel Carrier Preferences](#)
- [Defining a Node's Relationships](#)
- [Defining a Node's LTL Carrier Preferences](#)

3.1.2.5.1 Defining a Node's Primary Information

A node's primary information determines how it is identified throughout the system.

To define a node's primary information:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the Primary Info tab.
3. Enter information in the applicable fields. Refer to [Table 3–19](#) for field value descriptions.
4. Choose .

The screenshot displays the 'Node Primary Info Tab' with the following sections:

- Roles:** Buyer, Carrier, Enterprise, Hub, Node (checked), Seller.
- Enterprises:** MS, MS-INW-ORG, SS. Primary Enterprise: DEFAULT.
- Node Attributes:**
 - GLN: []
 - Node Type: Distribution Center
 - Identified By Parent As: FZ
 - Agent Criteria Group: []
 - Export License Number: []
 - License Expiration Date: 02/06/2006
 - BOL Prefix: 00
 - Default Declared Value: 0
 - Third Party Logistics Node
 - Maintain Inventory Cost
 - Requires Serial Number Tracking
- Execution In Node Using:**
 - External Application
 - Yantra Application Consoles
 - Yantra Networked WMS
 - Yantra WMS 6.2
- Action Name:** []
- Primary Info Panel:** Sourcing/Scheduling, Parcel Carrier Preferences, Relationship From Nodes, Relationship To Nodes, Print Format Preferences, LTL Carrier Preferences.

Table 3–19 Node Primary Info Tab

Field	Description
Node Type	
GLN	Enter the global location number.
Node Type	Select the node type for this node from the drop-down list.
Identified By Parent As	Enter the name the node's parent uses to identify it.
Agent Criteria Group	Select an agent criteria group from the drop-down list.
Export License Number	Enter the node's license number used for export shipments.
License Expiration Date	Enter the date the export license expires.
BOL Prefix	Enter the label this node uses as a prefix on the bills of lading it creates, if applicable.

Table 3–19 Node Primary Info Tab

Field	Description
Default Declared Value	Enter the price to appear as the default for Declared Value in the Application Consoles. This price is typically used by parcel carriers for computing insurance.
Third Party Logistics Node	Choose this if the node is part of a third-party logistics business model. Chained orders are not created for nodes marked as a third-party logistics node.
Maintain Inventory Cost	Choose this if the node maintains its own inventory costs. When this option is selected, cost must be entered for each inventory adjustment that happens at this node. If you choose this, then inventory adjustments made for this node must be approved. Adjustments awaiting approval are called pending adjustments. The actual adjustment does not occur until the pending adjustment has been approved.
Requires Serial Number Tracking	Choose this if the node requires inventory to be tracked using serial numbers.
Execution in Node Using	
External Application	Choose this to have order releases interface through events. For more information about events, see Section 4.2.6, "Defining Transactions" on page 158.
Yantra Application Consoles	Choose this to have order releases interface through the Yantra 7x Application Consoles.
Yantra Networked WMS	Choose this to have order releases interface through a Yantra 7x WMS.

Table 3–19 Node Primary Info Tab

Field	Description
Yantra WMS 6.2	Choose this to have order releases interface through a Yantra WMS version prior to and including Release 6.2.
Action Name	<p>If you chose External Application, select the action to associate with it. For more information about actions, see Section 4.2.9, "Defining Actions" on page 201.</p> <p>Note: In the drop-down, only the Actions linked to the Primary Enterprise of this Node/Organization will be available. Any actions created from this screen using the + button will be linked to the Primary Enterprise of the User's Organization. As a result, they may not be available for the Node/Organization being created.</p> <p>In a multi-enterprise environment, ensure that Actions are created for the appropriate Enterprises first (when logged in as that Enterprise user). Subsequently, mapping of nodes to actions can be done when logged in either as an Enterprise user or as a Hub user.</p>

3.1.2.5.2 Defining a Node's Sourcing and Scheduling

Setting up a node's sourcing and scheduling enables you to define how orders are sourced from this node.

To define node sourcing and scheduling:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the Sourcing/Scheduling tab.
3. Enter information in the applicable fields. Refer to [Table 3–20](#) for field value descriptions.

Node Attributes **Advanced Attributes**

Node needs to be notified at least hours prior to expected time of shipment

Release an order to this node days before expected time of shipment

Do not schedule an order to this node more than days before expected time of shipment

Procure To Ship Allowed Requires Transfer Acceptance

Work Order Creation Is Allowed Substitution Is Allowed

Receipt Processing Time (Hours) Use End Of Shift Time

Receiving Calendar Shipping Calendar

Node Can Ship To

Any Address

All Nodes Specific Nodes Of Type

Regional DC Store

Primary Info

Sourcing/Scheduling

Parcel Carrier Preferences

Relationship From Nodes

Relationship To Nodes

Print Format Preferences

LTL Carrier Preferences

Table 3–20 Node Sourcing/Scheduling Tab

Field	Description
Node needs to be notified at least <i>n</i> hours prior to expected time of shipment	Enter the number of hours before the expected shipment date that communication must be sent to the node to ship an order.
Release an order to this node <i>n</i> days before expected time of shipment	Enter the minimum number of days it takes to ship an order once it has been released to the node.
Do not schedule an order to this node more than <i>n</i> days before expected time of shipment	Enter the maximum number of business days that a schedule for an order can be sent to a node for it to be fulfilled. This number is used when performing earliest schedule date calculations. This parameter is only considered if the node is pre-specified on the order line.

Table 3–20 Node Sourcing/Scheduling Tab

Field	Description
Procure To Ship Allowed	<p>Choose this if the node can accept procurement chained orders. A procurement chained order is a type of order that is created when a node has to source inventory from another node. A chained order is an order that is linked to a parent order in which the lifecycle of one effects the other. There are two types of procurement chained orders: procurement transfer orders and procurement purchase orders.</p> <p>A transfer order is a type of chained order that is created when the node you are configuring needs to replenish their stock from another node within the organization to fulfill an order.</p> <p>A procurement purchase order is a type of chained order that is created when the node you are configuring needs to replenish their stock from another node that belongs to an external organization to fulfill an order.</p> <p>When setting up procurement from one node to another, you must define the billing address of each node. Billing addresses are defined in the Payment Info panel of the Organization Details screen. Also, a legal entity must be present in the organization hierarchy for the procured-from ship node.</p>
Requires Transfer Acceptance	<p>Choose this if you want this node to accept a procurement to confirm availability before proceeding with the order.</p>
Work Order Creation Is Allowed	<p>Choose this if you want to use Work Orders to support compliance services at this node. Work Orders describe the service activities to customize items based on a buyer's requests.</p>
Substitution Is Allowed	<p>Choose this if substitution of product items within an order is allowed.</p>
Receipt Processing Time (Hours)	<p>Enter the how many hours it takes the node to process receipts.</p>

Table 3–20 Node Sourcing/Scheduling Tab

Field	Description
Use End Of Shift Time	<p>Select this field if you want the node to base shipment time by the end of the next feasible shift.</p> <p>Unselect this field if you want the node to base shipment time by any given node parameters, such as Minimum Notification Time, and the time a shipment can actually be shipped.</p> <p>For example, a node works five days a week, with two shifts, 8AM - 4PM and 4PM - 8PM.</p> <p>The node's Minimum Notification Time is set to 2 hours.</p> <p>If an order is sent to a node at Friday 1PM, the order is scheduled to ship on same day at 4PM if Use End Of Shift Time is set to Y. The order is scheduled to ship on same day at 3PM if Use End Of Shift Time is set to N.</p> <p>If an order is sent to a node at Friday 3PM, the order scheduled to ship on same day at 8PM if Use End Of Shift Time is set to Y. The order is scheduled to ship on same day at 5PM if Use End Of Shift Time is set to N.</p> <p>Use End Of Shift Time is only applicable for product lines.</p>
Receiving Calendar	<p>Select the calendar to use to determine the available shifts for receiving deliveries at the node. The calendars of the node as well as the calendars of the primary enterprise of the node display in this drop-down list.</p>
Shipping Calendar	<p>Select the calendar to use to determine the available shifts for shipping from the node. The calendars of the node as well as the calendars of the primary enterprise of the node display in this drop-down list.</p>
<p>Node Can Ship To</p> <p>Select the node types that this node can ship to. The availability of these checkboxes depends on what node types have been defined, except the Any Address checkbox which is always available.</p>	
Any Address	<p>Check this checkbox to allow this node to ship to any address.</p>

Table 3–20 Node Sourcing/Scheduling Tab

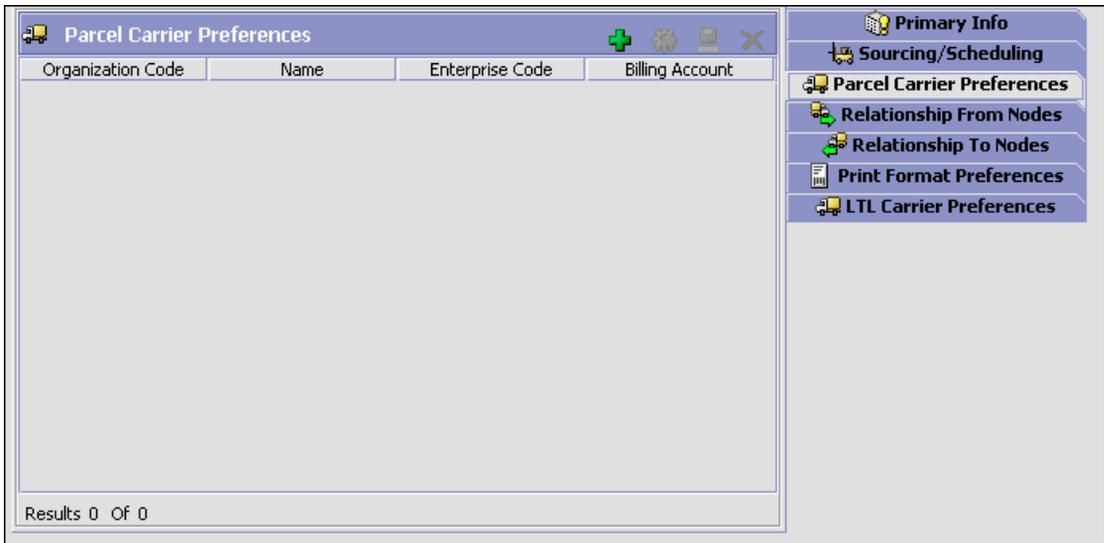
Field	Description
All Nodes	Select this radio button to allow this node to ship to all nodes.
Specific Nodes Of Type	Select this radio button and check the desired available checkboxes to allow this node to ship only to nodes of a specific node type.

3.1.2.5.3 Defining a Node’s Parcel Carrier Preferences

You can identify the carriers a node uses and define how they should interact. You can add, modify, and delete carrier preferences.

To add parcel carrier preferences to a node:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the Parcel Carrier Preferences tab.



3. Choose . The Parcel Carrier Preferences Detail pop-up window displays.

4. Enter information in the applicable fields. Refer to [Table 3–21](#) for field value descriptions.
5. Choose .

Table 3–21 Parcel Carrier Preferences Detail Pop-up Window

Field	Description
Carrier	Choose the carrier to set up the preferences. This is a mandatory field.
Enterprise Code	Choose the enterprise code.
Shipping Account	Enter the shipping account information.
Should shipping charges be billed to third party organization?	Check this box if shipping charges are to be billed to a third party organization.
Third Party Organization Details	

Field	Description
Organization Code	Choose the organization code of the third party organization.
Billing Account	Choose the billing account information of the third party organization.
Online Control #	Enter the online control number for the carrier preference.
International Shipping Integration Required	Check this box if international shipping integration is required.
Domestic Shipping Integration Required	Check this box if domestic shipping integration is required.
Allow system to automatically open a manifest	Check this box if the system should be allowed to automatically open a manifest.

To modify a node's parcel carrier preferences:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the Parcel Carrier Preferences tab.
3. Select the applicable carrier preference and choose . The Parcel Carrier Preferences Detail pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–21](#) for field value descriptions.
5. Choose .

To delete a node's parcel carrier preferences:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the Parcel Carrier Preferences tab.
3. Select the applicable carrier preference and choose .

3.1.2.5.4 Defining a Node's Relationships

A transfer order is a type of chained order that is created when a node that belongs to the organization you are configuring needs to replenish

their stock from another node within the organization to fulfill an order. A chained order is an order that is linked to a parent order in which the lifecycle of one effects the other.

You can define a relationship between the node you are defining and another node. Within this relationship you can define a transfer schedule, including the transit time to procure items from a node, on a day-of-week basis. The schedule is used for calculating expected dates.

You can define a transfer schedule that determines when items can be shipped from one node to another, including the transit time to procure items from a node, on a day-of-week basis. The schedule is used for calculating expected dates.

You can create, modify, and delete relationships.

Creating a Node Relationship

To create a node relationship:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. To create a node relationship from the node to another node, choose the Relationship To Node tab. To create a node relationship to the node from another node, choose the Relationship To Node tab.
3. Choose . The Relationship Details pop-up window appears.
4. Enter information into the applicable fields. Refer to [Table 3–22](#) for field value descriptions.
5. Choose .

Table 3–22 Relationship Details Pop-Up Window

Field	Description
Relationship Type	Select a relationship type for this relationship from the drop-down list.
From Node	Select the node from which items can be procured. For the Relationship To Node tab, this option is defaulted to the node you are configuring and disabled.
To Node	Select the node to which transfer order items are sent. For the Relationship From Node tab, this option is defaulted to the node you are configuring and disabled.
Transfer Schedule exists	Check this box if a transfer schedule exists between the From Node and To Node for this relationship. If this box is not checked, additional transfer schedule configuration is disabled.

Table 3–22 Relationship Details Pop-Up Window

Field	Description
Default Transit Days	Enter how many days it takes to deliver a transfer order to the node in To Node after it has been shipped from the node in From Node. This field is used as the default if an Override Transit Days is not entered (or has a value of 0) for a Shipped On day. Valid values are 0 through 999,999. If no value is entered, a 0 will be filled in automatically.
Shipped on	
Day of Week	Select the days that the node in From Node is available to ship procurement items to the node in To Node.
Override Transit Days	Enter the transit time that overrides the value entered in Default Transit Days for the selected day of the week, if applicable. Valid values are 0 through 999,999. If no value is entered, a 0 will be filled in automatically.

Modifying a Node Relationship

To modify a node relationship:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. To modify a node relationship from the node to another node, choose the Relationship To Node tab. To modify a node relationship to the node from another node, choose the Relationship To Node tab.
3. From the table, locate the applicable relationship and choose . The Relationship Details pop-up window appears.
4. Enter information into the applicable fields. Refer to [Table 3–22](#) for field value descriptions.
5. Choose .

Deleting a Node Relationship

To delete a node relationship:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.

2. To delete a node relationship from the node to another node, choose the Relationship To Node tab. To modify a node relationship to the node from another node, choose the Relationship To Node tab.
3. From the table, locate the applicable relationship choose .

3.1.2.5.5 Defining a Node's LTL Carrier Preferences

You can define the PRO Number Generation settings for the carriers a node uses and define how they should be used.

PRO Number refers to the unique progressive or serial number assigned by the carrier to identify and track a specific shipment. This is used on freight bills, bills of lading, and waybills for invoicing and tracking purposes.

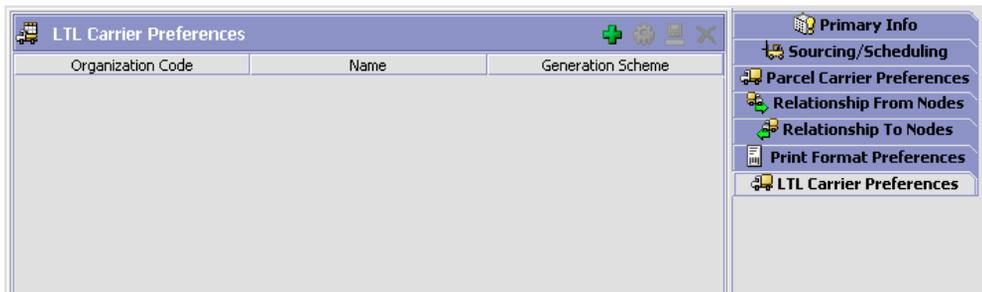
A warehouse may define the range of PRO Numbers assigned by a carrier. The PRO Number has a fixed length defined for each carrier, and may contain a prefix.

PRO Number is typically generated during routing for a load with carrier type 'LTL'. The PRO Number is regenerated automatically when the carrier/service on the load is changed.

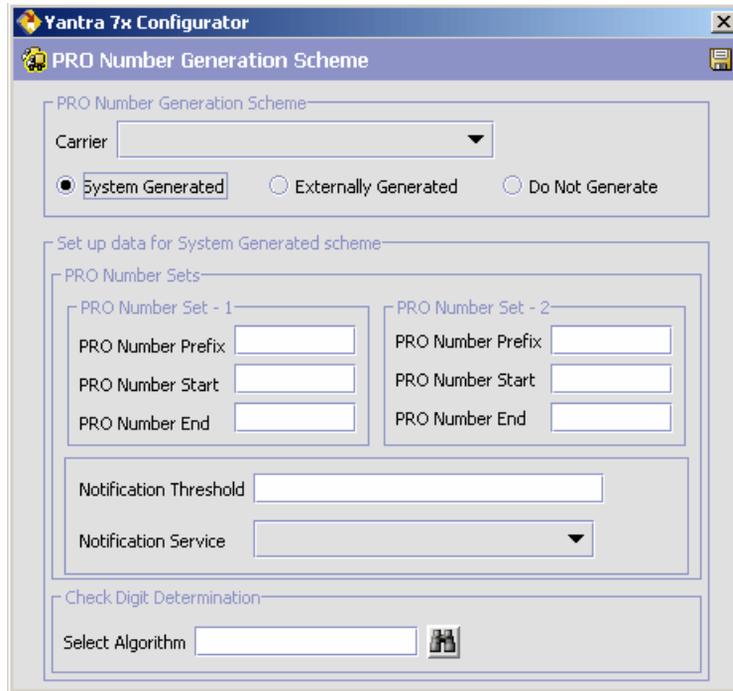
You can add, modify, and delete LTL carrier preferences.

To add LTL carrier preferences to a node:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the LTL Carrier Preferences tab.



3. Choose . The PRO Number Generation Scheme pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–23](#) for field value descriptions.
5. Choose .



The screenshot shows the 'PRO Number Generation Scheme' dialog box within the 'Yantra 7x Configurator' application. The dialog is divided into several sections:

- PRO Number Generation Scheme:** Contains a 'Carrier' dropdown menu and three radio buttons: 'System Generated' (selected), 'Externally Generated', and 'Do Not Generate'.
- Set up data for System Generated scheme:** This section is further divided into:
 - PRO Number Sets:** Two columns of input fields. The first column is for 'PRO Number Set - 1' and the second for 'PRO Number Set - 2'. Each column contains fields for 'PRO Number Prefix', 'PRO Number Start', and 'PRO Number End'.
 - Notification Threshold:** A single-line text input field.
 - Notification Service:** A dropdown menu.
- Check Digit Determination:** Contains a 'Select Algorithm' dropdown menu and a small icon.

Table 3–23 PRO Number Generation Scheme Pop-Up Window

Field	Description
PRO Number Generation Scheme	<p>One of the following actions is initiated based on the Generation Scheme selected:</p> <ul style="list-style-type: none"> • Yantra 7x WMS generates the PRO Numbers • Yantra 7x WMS uses the PRO Numbers generated by an external system, or • Yantra 7x WMS does not generate PRO Numbers.
Carrier	Choose the carrier whose PRO Number Generation Scheme is being defined.
System Generated	Select 'System Generated' if Yantra 7x WMS should generate PRO Numbers.
Externally Generated	Select 'Externally Generated' if Yantra 7x WMS should use PRO Numbers generated by an external system.
Do Not Generate	Select 'Do Not Generate' if PRO Numbers should not be generated.
Set up data for System Generated scheme	This is applicable only when Yantra 7x WMS generates the PRO Numbers.
PRO Number Sets	When the first range of PRO Numbers is used up, Yantra 7x WMS utilizes the second range of PRO Numbers, and vice versa.
PRO Number Set - 1	Defines the first range of PRO Numbers to be used by Yantra 7x WMS.
PRO Number Prefix	<p>Enter the PRO Number Prefix to be used in generating the first range of PRO Numbers.</p> <p>PRO Number Prefix can be alpha numeric, and is not included for computing the check digit.</p>
PRO Number Start	<p>Enter the PRO Number Start to be used in generating the first range of PRO Numbers.</p> <p>PRO Number Start is the starting number for the first set of PRO Numbers.</p>
PRO Number End	<p>Enter the PRO Number End to be used in generating the first range of PRO number.</p> <p>PRO Number End is the ending number for the first set of PRO Numbers.</p>

Table 3–23 PRO Number Generation Scheme Pop-Up Window

Field	Description
PRO Number Set - 2	Defines the second range of PRO Numbers to be used by Yantra 7x WMS.
PRO Number Prefix	Enter the PRO Number Prefix to be used in generating the second range of PRO Numbers. PRO Number Prefix can be alpha numeric, and is not included for computing the check digit.
PRO Number Start	Enter the PRO Number Start to be used in generating the second range of PRO Numbers. PRO Number Start is the starting number for the second set of PRO Numbers.
PRO Number End	Enter the PRO Number End to be used in generating the second range of PRO Numbers. PRO Number End is the ending number for the second set of PRO Numbers.
Notification Threshold	Enter the notification threshold at which an alert is raised to the user. This gains importance in instances where the second set of PRO Numbers is not defined. The notification threshold defines the number of unassigned PRO Numbers available till the PRO Number End. This enables the warehouse to talk to the carrier and get a new range of PRO Numbers. Note: If the PRO Number End is X, and the notification threshold is set to 50, Yantra 7x WMS raises an alert to the user, when the current PRO Number (unassigned) reaches X-50.
Notification Service	Choose Notification Service to be used. Notification Service is the service that will be invoked when Notification Threshold is reached.

Table 3–23 PRO Number Generation Scheme Pop-Up Window

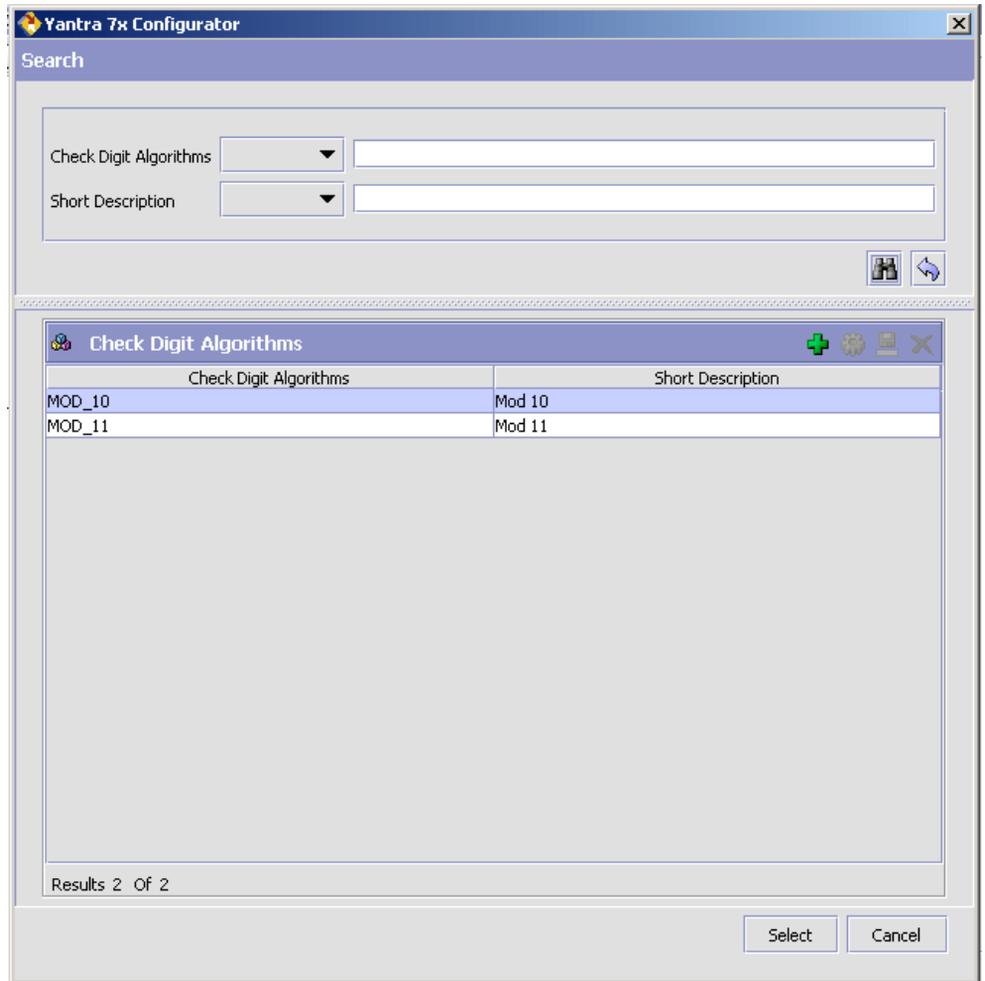
Field	Description
Check Digit Determination	Defines the check digit algorithm to be used for PRO Generation.
Select Algorithm	<p>Choose the relevant algorithm for check digit determination.</p> <p>Typical values are mod-11 and mod-10.</p> <p>Note: When 'mod-11' or 'mod-10' schema of check digit determination is chosen, the check digit is generated out-of-the-box by Yantra 7x WMS.</p> <p>For more details about Check Digit Determination Algorithm, see Setting Up a Check Digit Determination Algorithm.</p>

Setting Up a Check Digit Determination Algorithm

Creating a Check Digit Determination Algorithm

To set up a check digit determination algorithm:

1. In the PRO Number Generation Scheme pop-up window, choose .
2. The Check Digit Algorithms Search window appears.



3. In the Check Digit Algorithms panel, choose . The Check Digit Logic Details pop-up window appears.
4. Enter information in the applicable fields. Refer [Table 3–24](#) for field value descriptions.
5. Choose .

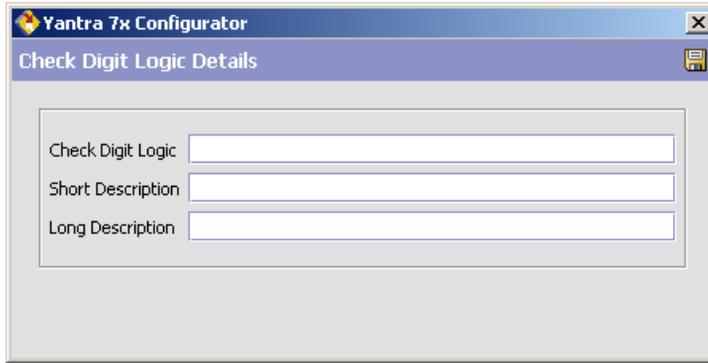


Table 3–24 *Check Digit Logic Details Popup Window*

Field	Description
Check Digit Logic	Enter the check digit logic for the algorithm being created.
Short Description	Enter a short description for the algorithm being created.
Long Description	Enter a long description for the algorithm being created.

Creating a New Check Digit Determination Algorithm from an Existing Check Digit Determination Algorithm

To create a new check digit determination algorithm from an existing check digit determination algorithm:

1. In the PRO Number Generation Scheme pop-up window, choose .
2. The Check Digit Algorithms Search window appears.
3. In the Search panel, enter applicable search criteria, and choose .
4. The relevant search results are displayed in the Check Digit Algorithms panel.
5. From the Check Digit Algorithms list, choose the Check Digit Algorithm to be copied from.
6. Choose . The Check Digit Logic Details pop-up window appears.

7. Enter information in the applicable fields. Refer [Table 3–24](#) for field value descriptions.
8. Choose .

Modifying a Check Digit Determination Algorithm

Once a check digit determination algorithm has been created, it can be modified.

To modify a check digit determination algorithm:

1. In the PRO Number Generation Scheme pop-up window, choose .
2. The Check Digit Algorithms Search window appears.
3. In the Search panel, enter applicable search criteria, and choose .
4. The relevant search results are displayed in the Check Digit Algorithms panel.
5. From the Check Digit Algorithms list, choose the Check Digit Algorithm to be modified.
6. Choose . The Check Digit Logic Details pop-up window appears.
7. Enter information in the applicable fields. Refer [Table 3–24](#) for field value descriptions.
8. Choose .

Deleting a Check Digit Determination Algorithm

To delete a check digit determination algorithm:

1. In the PRO Number Generation Scheme pop-up window, choose .
2. The Check Digit Algorithms Search window appears.
3. In the Search panel, enter applicable search criteria, and choose .
4. The relevant search results are displayed in the Check Digit Algorithms panel.
5. From the Check Digit Algorithms list, choose the Check Digit Algorithm to be deleted.
6. Choose .

To modify a node's LTL carrier preferences:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the LTL Carrier Preferences tab.
3. Select the applicable LTL Carrier Preference and choose . The PRO Number Generation Scheme pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–23](#) for field value descriptions.
5. Choose .

To delete a node's LTL carrier preferences:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the LTL Carrier Preferences tab.
3. Select the applicable LTL Carrier Preference and choose .

3.1.2.6 Defining a Node's Advanced Inventory Attributes

You can determine if a node's inventory is maintained within Yantra 7x. You can also determine if the node you are configuring is itself an inventory node or identifies the inventory node to which it belongs.

To define a node's advanced inventory attributes:

1. From the Roles & Participation tab in the Organization Details window, choose Advanced Attributes.
2. Enter information in the applicable fields. Refer to [Table 3–25](#) for field value descriptions.
3. Choose .

Inbound Compliance **Node Attributes** **Advanced Attributes**

Inventory Information Not Available To Yantra

Inventory Is Made Available To Yantra

This Organization Is An Inventory Organization Get External Inventory Real Time

Inventory Organization Is **DEFAULT** ▼

Capture Tag Information

Only when receiving

When performing any node operation

Only when shipping, if the buyer mandates

Table 3–25 Advanced Attributes, Inventory Tab

Field	Description
Inventory Information Not Available To Yantra	Select this option if the node uses an inventory application other than Yantra 7x to track inventory. If this option is selected, Yantra 7x never has access to the inventory picture for this node, and never attempts to do any inventory availability checks.
Inventory Is Made Available To Yantra	Select this if the node tracks inventory using either Yantra 7x or another inventory application. The node's inventory picture can be used in Yantra 7x for availability calculation, inventory tracking, and distribution. If you are configuring a node, this field determines if inventory is tracked or infinite at the node.
Get External Inventory Real Time	Select this field if the inventory is available to Yantra 7x, but comes from an external system instead of being maintained internally. The inventory picture will then be obtained by calling the <code>getExternalInventory</code> user exit. Note: This checkbox is only available when the inventory information is available to Yantra 7x, and the organization plays a role other than a node.

Table 3–25 Advanced Attributes, Inventory Tab

Field	Description
Get External Supply Real Time	Select this field if the supply is available to Yantra 7x, but comes from an external system instead of being maintained internally. The supply picture is obtained by calling the <code>getExternalSupply</code> user exit. Note: This checkbox is only available when the inventory information is available to Yantra 7x, and the organization is a node only.
Capture Tag Information	
Only when receiving	Select this option to capture an item's tag information only when receiving inventory.
When performing any node operation	Select this option if you want to capture an item's tag information when performing any node operation, such as receiving, picking, putaway, or counting.
Only when shipping, if the buyer mandates	Select this option if you want to capture an item's tag information only when shipping inventory to a buyer who mandates it.

3.1.2.7 Defining an Organization's Advanced Inventory Attributes

You can determine if an organization's inventory is maintained within Yantra 7x. You can also determine if the organization you are configuring is an inventory organization or the inventory organization it belongs to.

Important: This is an installation level configuration only. Do not attempt to reconfigure the parameters on this tab mid-implementation.

Note: When creating an organization through the save as operation, the new organization's inventory organization will be the inventory organization of the source organization. If the source organization is its own inventory organization, the source organization will be set as the inventory organization of the new organization.

To define an organization's advanced inventory attributes:

1. From the Roles & Participation tab in the Organization Details window, choose Advanced Attributes.
2. Choose the Inventory tab.
3. Choose .

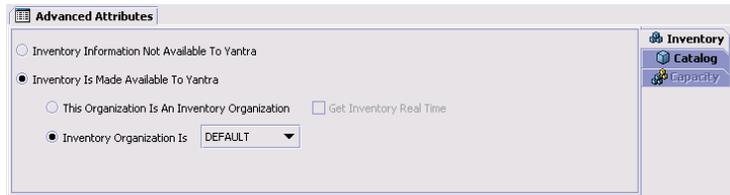


Table 3–26 Inventory Tab

Field	Description
Inventory Information Not Available To Yantra	<p>Select this option if the organization uses an inventory application other than Yantra 7x to track supply and demand.</p> <p>If this option is selected, Yantra 7x will never have access to the inventory picture for this organization, and will never attempt to do any inventory availability checks. Orders can still be placed using this organization as a seller, but the orders will be scheduled without any inventory availability checks.</p>
Inventory Is Made Available To Yantra	<p>Select this if the organization tracks inventory supply and demand using either Yantra 7x or another inventory application. The organization's supply and demand picture can be used in Yantra 7x for availability calculation, inventory tracking, and distribution.</p> <p>If you are configuring a node, this field determines if inventory is tracked or infinite at the node.</p>
This Organization Is An Inventory Organization	<p>Select this if you selected Inventory Is Maintained Within Yantra 7x and you want to specify this organization as an inventory consolidator.</p> <p>An inventory organization:</p> <ul style="list-style-type: none"> • Provides inventory identification for a product. For example, different organizations can have different product identification IDs for the same inventory item. The inventory organization provides a mechanism to rationalize these product IDs into a single nomenclature across multiple organizations. • Establishes ownership of inventory when a single physical location is shared across multiple organizations without having to create multiple logical locations to establish the inventory ownership. • Provides inventory separation, allowing all organizations that are part of the inventory organization to have visibility to the inventory of all of the other organizations that are part of the inventory organization.

Table 3–26 Inventory Tab

Field	Description
Inventory Organization Is	<p>Select this and the applicable inventory organization if you selected Inventory Is Made Available To Yantra and you want to associate this organization as part of the applicable inventory organization.</p> <p>Important: The organization should have the same catalog organization as the inventory organization you are associating with.</p>
Get Inventory Real Time	<p>Select this field if the inventory is available to Yantra 7x, but comes from an external system instead of being maintained internally. The inventory picture will then be obtained by calling the <code>getExternalInventory</code> user exit.</p>
Get External Supply Real Time	<p>Select this field if the supply is available to Yantra 7x, but comes from an external system instead of being maintained internally. The supply picture will then be obtained by calling the <code>getExternalSupply</code> user exit.</p> <p>Note: This checkbox is only available when the organization has been defined only as a node.</p>

3.1.2.8 Defining an Organization’s Advanced Catalog Attributes

You can determine if an organization maintains its own catalog or if it is maintained by another organization.

Important: This is an installation level configuration only. Do not attempt to reconfigure the parameters on this tab mid-implementation.

Note: When creating an organization through the save as operation, the new organization's catalog organization will be the catalog organization of the source organization. If the source organization is its own catalog organization, the source organization will be set as the catalog organization of the new organization.

To define an organization's advanced catalog attributes:

1. From the Roles & Participation tab in the Organization Details window, choose Advanced Attributes.
2. Choose the Catalog tab.
3. Enter information in the applicable fields. Refer to [Table 3–27](#) for field value descriptions.
4. Choose .



The screenshot shows a software interface with a tabbed menu at the top containing 'Carrier Attributes', 'Enterprise Attributes', 'Seller Attributes', and 'Advanced Attributes'. The 'Advanced Attributes' tab is active. Below the tabs, there are two radio button options: 'Organization Defines Its Own Catalog' (which is selected) and 'Catalog Defined By' (which has a dropdown arrow). To the right of the main content area, there is a vertical sidebar with three sub-tabs: 'Inventory', 'Catalog', and 'Capacity'.

Table 3–27 Catalog Tab

Field	Description
Organization Defines Its Own Catalog	Select this field in the organization defines it's own item master catalog. The item master that this organization defines can be shared with other organizations.
Catalog Defined By	Select this field and select the applicable catalog organization if you want to use the organization's item master catalog without having to create a separate catalog of your own.

3.1.2.9 Defining an Organization's Advanced Capacity Attributes

You can determine if an organization maintains its own capacity or if it is maintained by another organization.

Important: This is an installation level configuration only. Do not attempt to reconfigure the parameters on this tab mid-implementation.

Note: When creating an organization through the save as operation, the new organization's capacity organization will be the capacity organization of the source organization. If the source organization is its own capacity organization, the source organization will be set as the capacity organization of the new organization.

To define an organization's advanced capacity attributes:

1. From the Roles & Participation tab in the Organization Details window, choose Advanced Attributes.
2. Choose the Catalog tab.
3. Enter information in the applicable fields. Refer to [Table 3–28](#) for field value descriptions.
4. Choose .

Table 3–28 Capacity Tab

Field	Description
This Organization Is A Capacity Organization	<p>Select This Organization Is A Capacity Organization if you want to specify this organization as an service capacity consolidator.</p> <p>A capacity organization:</p> <ul style="list-style-type: none"> • Provides service capacity identification for a product. • Establishes ownership of capacity when a single physical location is shared across multiple organizations without having to create multiple logical locations to establish the capacity ownership. • Provides capacity separation, allowing all organizations that are part of the capacity organization to have visibility to the capacities of all of the other organizations that are part of the same capacity organization.

Table 3–28 Capacity Tab

Field	Description
Capacity Organization Is	Select Capacity Organization Is and select the applicable capacity organization if you want to associate this organization as part of the applicable capacity organization. Important: The organization should have the same catalog organization as the capacity organization you are associating with.
Get Capacity Real Time	Select this field if the organization makes capacity information available to Yantra 7x from an external system. When Yantra 7x checks for capacity, the external system provides the capacity in response to a user exit call from Yantra 7x.

3.1.3 Defining Communication Protocols

Communication protocols are the means by which an organization communicates in the Hub environment. For example, if the organization you are setting up uses both an FTP site and e-mail services to send and receive documents, you would identify them along with their protocols here. You can create, modify, and delete communication protocols.

You can use the Communication tab for:

- [Creating a Communication Protocol](#)
- [Modifying a Communication Protocol](#)
- [Deleting a Communication Protocol](#)

3.1.3.1 Creating a Communication Protocol

To create a communication protocol:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists are displayed.
2. From the Communication Protocols list, select . The Communication Protocol Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 3–29](#) for field value descriptions.
4. Choose .

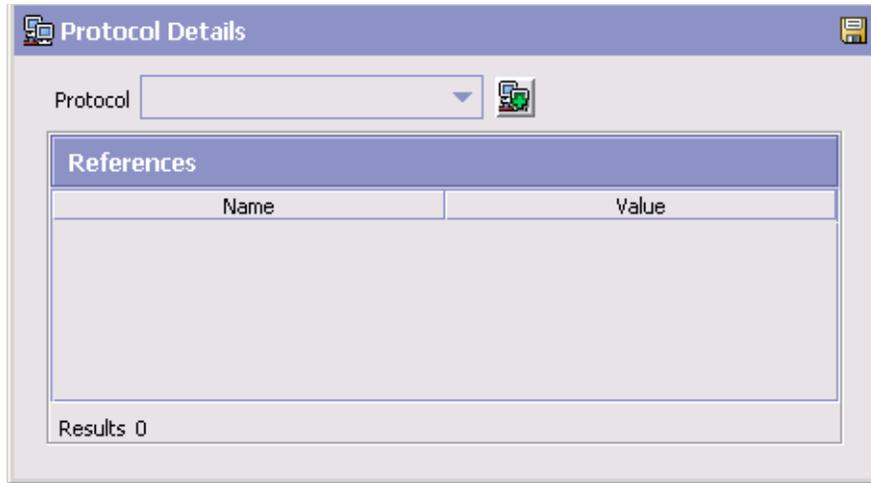


Table 3–29 Communication Protocol Details Pop-Up Window

Field	Description
Protocol	Select the type of protocol you want to set up. For more information about protocols, see Section 10.1, "Defining Protocol Codes" on page 327. You must choose  before entering any references.
References	
Name	Enter a reference name for the protocol. For example, IP address.
Value	Enter the value for the reference name.

3.1.3.2 Modifying a Communication Protocol

To modify a communication protocol:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents list displays.
2. From the Communication Protocols list, select the applicable communication protocol and choose . The Communication Protocol Details pop-up window displays.
3. Modify information in the applicable fields. Refer to [Table 3–29](#) for field value descriptions.

4. Choose .

3.1.3.3 Deleting a Communication Protocol

To delete a communication protocol:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists are displayed.
2. From the Communication Protocols list, select the applicable communication protocol and choose .

3.1.4 Defining an Organization's Payment Information

An organization that makes any type of monetary transactions with other organizations must have payment information set up. This information provides all parties with an account number, billing address, and tax information.

To set up an organization's payment information:

1. In the Organization Details window, choose the Payment Info tab.
2. Enter information in the applicable fields. Refer to [Table 3–30](#) for field value descriptions.
3. Choose .

Table 3–30 *Payment Info Tab*

Field	Description
Account Number With Hub	Enter the organization’s account number used for monetary transactions with the Hub organization, if applicable.
Tax Information	
Tax Payer ID	Enter the organization’s tax payer identification number. This number identifies the organization as a tax paying entity.
Authority Type	Enter the authority type given for an exemption certificate, if applicable.
Tax Exempt	Check this box if the organization is exempt from paying taxes.
Exemption Certificate	Enter the identification number of the exemption certificate.

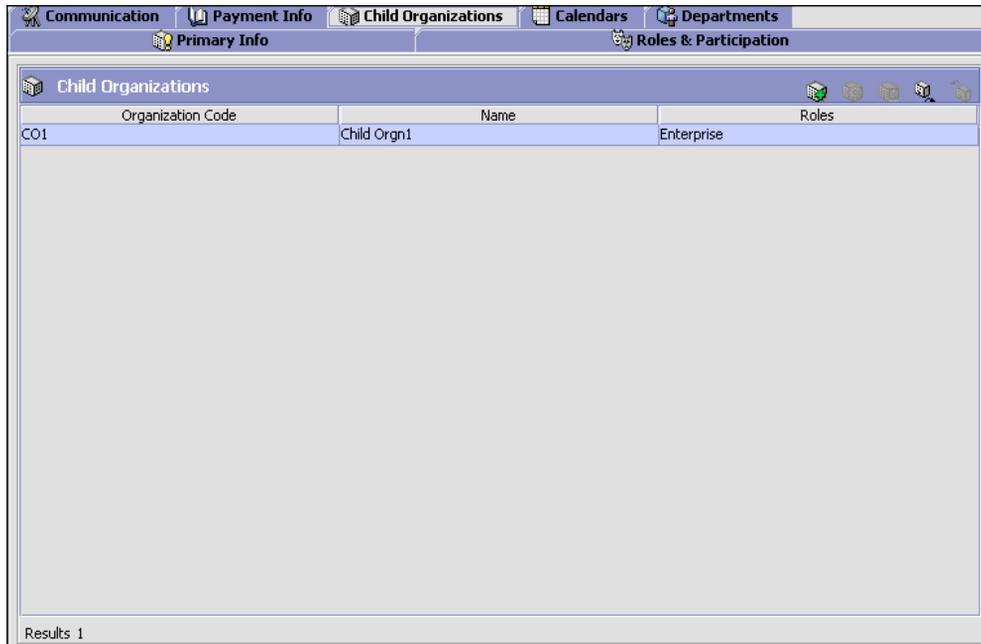
Table 3–30 Payment Info Tab

Field	Description
Issuing Authority	Enter the authority that issued the exemption certificate.
Tax Jurisdiction	Enter the tax jurisdiction that the exemption certificate was issued in.
Billing Address	<p>The the organization’s billing address. This information is mandatory.</p> <p>Choose  to enter an address.</p> <p>Choose the Contact Info tab to view additional contact information.</p>

3.1.5 Viewing an Organization’s Child Organizations

You can view any child organizations an organization may have.

To view an organization’s child organizations, choose the Child Organizations tab in the Organization Details window. You can create and modify organizations from this tab as described in detail in additional sections of this chapter.



3.1.6 Defining an Organization's Calendars

You can define an organization's working calendar. A working calendar is a span of dates for a defined period for which you can define any working shifts (for example, Day Shift, Night Shift), exception shifts (for example, extra shifts on the last day of the month for performing inventory stock), and exception days (for example, Fourth of July, New Years Day).

A node or an organization can choose its calendars as well as the calendars of its primary enterprise as its business calendar, shipping calendar, or receiving calendar.

A node or an organization can also inherit calendar definition from its primary enterprise when creating calendars. If a calendar is inherited from another calendar, the parent calendar's components such as Effective Periods, Shifts, Calendar Day Exceptions, and Exception Shifts can be used by the child calendar during runtime. This implies that the inherited calendars cannot specify their own effective periods or shifts. However, a child calendar has the ability to specify its own set of Calendar Day Exceptions and Exception Shifts. These are used in

conjunction with the parent calendar's components while retrieving the day details of the child calendar during runtime.

Note: The child calendar's Calendar Day Exceptions and Exception Shifts override those of the parent calendar if they fall on the same date.

The following limitations are assumed when inheriting calendars:

- A calendar of an organization or a node can only be inherited from a calendar of the primary enterprise and from its own calendar.
- The parent calendar cannot be an inherited calendar.
- An inherited calendar is not allowed to change to a non-inherited calendar and vice-versa.
- An inherited calendar is not allowed to specify its own effective periods and standard shifts.
- If a calendar is inherited from another calendar:
 - only Calendar Day Exceptions and Exception Shifts can be defined for the inherited calendar.
 - the exception dates must fall under one of the effective periods of the parent calendar. Moreover, the start time and end time of the exception shifts must match the start time and end time of a shift within that effective period.

To set up an organization's calendar:

1. In the Organization Details window, choose the Calendar tab. The Calendar list displays.
2. Select . The Create Calendar pop-up window displays.

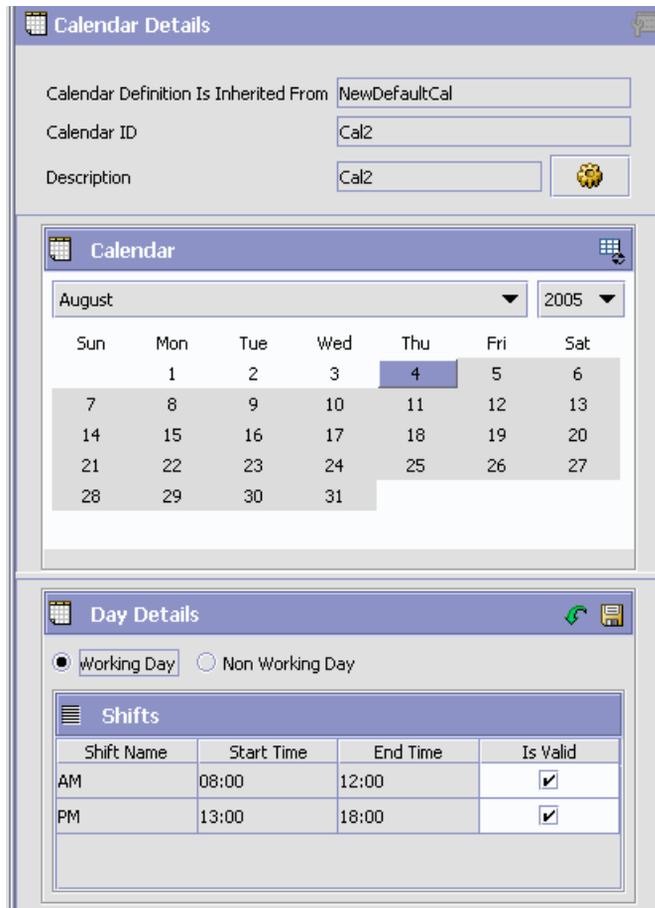
Table 3–31 Create Calendar Pop-up Window

Field	Description
Inherit Definition from Calendar	From the drop-down list, select the parent calendar from which you wish to inherit the calendar definition. Note: This list does not contain calendars that are inherited. Note: Once a calendar is selected from this list, the Default Effective From, Default Effective To, and Shifts for Effective Periods are disabled.
Calendar ID	Enter the identification for the calendar.
Description	Enter the description for the calendar.
Default Effective From	Enter the beginning date (according to the date format defined in the organization's locale) of the timeframe from which the calendar is effective.
Default Effective To	Enter the ending date (according to the date format defined in the organization's locale) of the timeframe to which the calendar is effective.
Shift Name	Enter the name of the shift.
Start Time	Enter the start time of the shift.
End Time	Enter the end time of the shift.

Select the days on which the shift is worked. Continue adding shifts as needed.

Note: A single calendar can have multiple effective periods.

3. Choose . The Calendar Details pop-up window displays.



Calendar Details

Calendar Definition Is Inherited From:

Calendar ID:

Description: 

Calendar

August 2005

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Day Details

Working Day Non Working Day

Shifts

Shift Name	Start Time	End Time	Is Valid
AM	08:00	12:00	<input checked="" type="checkbox"/>
PM	13:00	18:00	<input checked="" type="checkbox"/>

You can use the Calendar Details pop-up window for:

- [Defining a Calendar's Defaults](#)
- [Creating an Exception for a Particular Calendar Day](#)

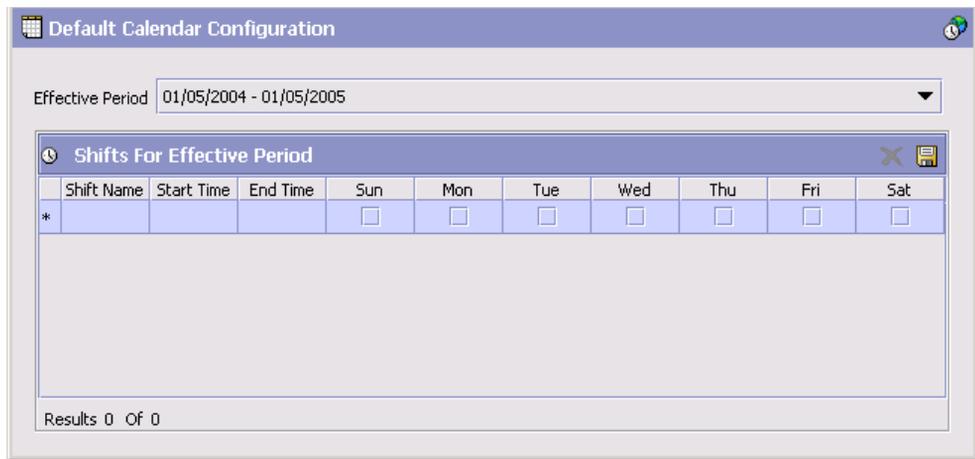
3.1.6.1 Defining a Calendar's Defaults

You can configure the default effective dates and working shifts for a calendar.

Note: The default effective dates and working shifts cannot be defined for inherited calendars.

To configure calendar defaults:

1. From the Calendar Details pop-up window, choose . The Default Calendar Configuration pop-up window appears.



2. From Effective Period, select the timeframe through which you want the calendar to be used.
 - Choose  to create additional effective periods to associate with the calendar.

Important: Effective periods cannot overlap each other. Nor can they start or end in the middle of the day.

3. In Shift Name, enter the name of the shift.
4. In Start Time, enter the time the shift starts.

5. In End Time, enter the time the shift ends.

Important: You cannot configure a shift to carry over to the next day.

6. From Shift Valid For, select the days of the week the shift you are configuring is valid for.
7. Choose . The shift now appears in Calendar Details pop-up window for any of the default days you have selected.

3.1.6.2 Creating an Exception for a Particular Calendar Day

You can mark a regular working day as a non-working day or indicate if certain shifts are valid or not for a particular day. The exceptions that you indicate are only valid for that day.

For example, if the Fourth of July is a holiday for the organization, and it falls on a Friday, which is a normal working day, you can mark that particular date as a non-working day.

As another example, if the organization has an extra shift for taking inventory on the last day of each month. You can create that shift and mark it as a valid shift for the last day of each month within your calendar's effective period.

Calendar Details

Calendar Definition Is Inherited From

Calendar ID

Description

Calendar

August 2005

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Day Details

Working Day
 Non Working Day

Shifts

Shift Name	Start Time	End Time	Is Valid
AM	08:00	12:00	<input checked="" type="checkbox"/>
PM	13:00	18:00	<input checked="" type="checkbox"/>

To mark a regular working day as a non-working day or vice versa, select the date you want from the Calendar Details pop-up window and choose the applicable exception from the Day Details frame.

To mark a default shift as valid or invalid:

1. From the Calendar Details window, select the date you want to work with.
2. If the date is a default non-work day:
 - a. Select Working Day.
 - b. Select the shift you want to mark as valid or invalid from the Shifts table. Check Is Valid to indicate that the shift is valid for

that particular date. Uncheck Is Valid to indicate that the shift is not valid for that particular date.

- c. Choose OK, then choose  from the Day Details window.
3. If the date is a default work day, select Non-Working Day. This automatically unchecks the Is Valid box for all shifts on that day. Choose  from the Day Details window.

Note: If you choose  from the Calendar Details window. The date that was highlighted shows the color of an exception day, even if no changes were made to that day.

4. You can apply the defaults for the overrides by choosing . But you cannot restore the defaults for parent overrides in inherited calendars.

3.1.7 Viewing an Organization's Departments

You can view all departments of an organization.

To view an organization's departments, choose the Departments tab in the Organization Details window.

Use the Departments tab for:

- [Creating a Department](#)
- [Modifying a Department](#)
- [Deleting a Department](#)

3.1.7.1 Creating a Department

To create a department:

1. In the Organization Details window, choose the Departments tab. The Department List displays.
2. From the Department List, select . The Department Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 3–32](#) for field value descriptions.
4. Choose .

The screenshot shows a window titled "Department Details". Inside the window, there are three text input fields stacked vertically. The first field is labeled "Department Code", the second is labeled "Department Name", and the third is labeled "Department Abbreviation". Each field is currently empty.

Table 3–32 *Department Details Pop-up Window*

Field	Description
Department Code	A unique code to identify the department.
Department Name	Name of the department.
Department Abbreviation	Abbreviation of the department name.

3.1.7.2 Modifying a Department

To modify a department:

1. In the Organization Details window, choose the Departments tab. The Department List displays.
2. From the Department List, select the applicable department and choose . The Department Details pop-up window displays.
3. Modify information in the applicable fields. Refer to [Table 3–32](#) for field value descriptions.
4. Choose .

3.1.7.3 Deleting a Department

To delete a department:

1. In the Organization Details window, choose the Departments tab. The Department List displays.

2. From the Department List, select the applicable department and choose .

3.1.8 Modifying an Organization

Once an organization has been created, you can modify it.

To modify an organization:

1. From the tree in the application rules side panel, choose Participant Modeling > Participant Setup. The Organization Search window appears in the work area.
2. Enter the applicable search criteria and choose . A list of organizations displays.
3. Select the organization and choose . The Organization Details window displays.
4. Refer to the topics under [Section 3.1, "Creating and Modifying an Organization"](#) for further instructions.

3.2 Creating and Modifying an Organizational Hierarchy

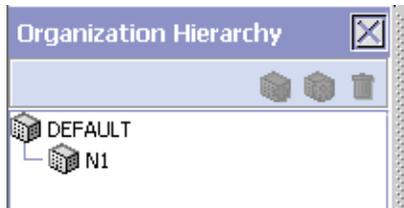
You can organize any existing organizations in an organizational hierarchy. This hierarchy can be used to configure the relationships between related organizations. For example, if you have a multi-divisional setting with one parent Hub and several Enterprises below it, you can organize all of the existing Enterprises under the Hub in the organizational hierarchy. You can add and remove organizations from the organizational hierarchy.

3.2.1 Creating an Organizational Hierarchy

To create an organizational hierarchy:

1. From the tree in the application rules side panel, choose Participant Modeling > Participant Setup. The Organization Search window appears in the work area.
2. Enter applicable search criteria and choose . A list of organizations displays.

3. Select the organization you want to build an organizational hierarchy for and choose .
4. The Organizational Hierarchy tree appears in the left frame with the name of the organization you chose. You can now add organizations to the hierarchy.



3.2.1.1 Adding an Organization to the Organizational Hierarchy

To add organizations to the organizational hierarchy:

1. From the tree select the organization you want to add the organization under.
1. From the Organizational Hierarchy tree, choose . The Add Organization to Hierarchy pop-up window appears.
2. Enter the applicable search criteria and choose . A list of organizations displays.
3. Select the organization you want to add to the organizational hierarchy and choose .

3.2.1.2 Removing an Organization from the Organizational Hierarchy

To remove an organization from the organizational hierarchy, select the organization you want to remove from the Organizational Hierarchy tree and choose .

3.3 Creating Node Types

You can create node types to classify nodes. You can use node types to define node relationships, and set inventory rules.

For more information on defining inventory node type rules, refer to the *Yantra 7x Inventory Synchronization Configuration Guide*.

You can use the Node Type branch for:

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

3.3.1 Creating a Node Type

To create a node type:

1. From the tree in the application rules side panel, choose Participant Modeling > Node Types. The Node Type window appears in the work area.
2. Choose . The Node Type Details pop-up window appears.
3. Enter information into the applicable fields. Refer to [Table 3–33](#) for field level descriptions.
4. Choose .

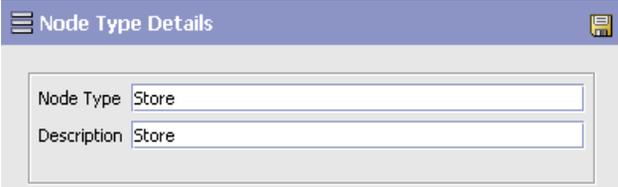


Table 3–33 Node Type Details Pop-up Window

Field	Description
Node Type	Enter a name for the node type.
Description	Enter a description for the node type.

3.3.2 Modifying a Node Type

To modify a node type:

1. From the tree in the application rules side panel, choose Participant Modeling > Node Types. The Node Type window appears in the work area.

2. Select the applicable node type and choose . The Node Type Details pop-up window appears.
3. Enter information into the applicable fields. Refer to [Table 3–33](#) for field level descriptions.
4. Choose .

3.3.3 Deleting a Node Type

To delete a node type:

1. From the tree in the application rules side panel, choose Participant Modeling > Node Types. The Node Type window appears in the work area.
2. Select the applicable node type and choose .

Configuring Process Models

Yantra 7x Process Modeling is the set up of Yantra 7x business process workflow. Yantra 7x workflow consists of the entire set of business logic that defines how Yantra 7x handles business documents and transactions on those documents. A **transaction** is a logical unit of work that encapsulates certain business logic. Transactions can be related to orders, inventory changes, returns, payment authorizations, or many other system events. Order Create, Inventory Monitor, and Send Release are examples of transactions.

Business process workflow consists of:

- Document types
- Repositories
- Process-type pipelines
- Transactions
- Conditions
- Events
- Statuses
- Actions
- Services

4.1 Document Type Configuration

Yantra 7x uses document types to carry information through a configured business process workflow. These documents are derived from base document types. A base document type defines the business documents

that Yantra 7x handles and defines a common storage structure for all derived document types.

The following base document types are defined in Yantra 7x:

- Order
- Load
- General
- Count
- Container
- Wave
- Work Order

Note: The available base document types are pre-defined and cannot be added to.

Document types are specific business documents that are derived from a base document type. For example, document types such as Sales Order and Purchase Order are derived from the Order base document type.

For detailed information on document types, refer to [Table E-1, "Document Types"](#) on page 657.

To complete a lifecycle, each document type has a set of different processes that it can go through. These processes are called process types. Every base document type has a defined set of process types in Yantra 7x.

The following are the process types defined in Yantra 7x (for the base document types):

- Order Fulfillment
- Order Negotiation
- Outbound Shipment
- Planned Order Execution
- Planned Order Negotiation
- Reverse Logistics

- Return Shipment
- Return Receipt
- Template Order
- Purchase Order Execution
- Purchase Order Negotiation
- Inbound Shipment
- Purchase Order Receipt
- Transfer Order Execution
- Transfer Order Delivery
- Transfer Order Receipt
- Load Execution
- General
- WMS Putaway
- WMS Layout Definition
- WMS Inventory
- Trailer Loading
- Task Execution
- Move Request Execution
- Manifesting
- Over Pack Build
- Count Execution
- Pack Process
- Outbound Picking
- VAS Process

Business rules such as payment collection rules and modification rules must be set up for each document type.

4.1.1 The Process Modeling Tree

In Yantra 7x, you can view a graphical representation of each base document type and its document and process types.

To view the process modeling tree:

1. From the tree in the application rules side panel, choose Process Modeling. The Process Modeling window appears in the work area.
2. Select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.

4.1.2 Creating a New Document Type

You may need to create a new document type if the rules pertaining to a key action, such as inventory updates, are affected. In this case, you can save an existing document type as a new custom document type. The new document type retains all of the process types associated with the document type you saved from. Database tables at both the document type level and the process type level are also copied to the new document type.

The following document type attributes are copied to the new document type:

- Document parameters
- Document templates
- Charge categories
- Charge names
- Common codes
- Order line types
- Purge criteria
- Business rules
- Receiving dispositions

The following process type level attributes are copied to the new document type:

- Process type rules

- Date types
- Process task types
- Statuses
- Status inventory types
- Modification types
- Modification rules
- Transactions
- Transaction pickup statuses
- Transaction drop statuses
- Events

To create a new document type:

1. From the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Document Types swimlane, right-click on the applicable document type and choose Save As. The New Document Type window appears.

The screenshot shows a 'New Document Type' configuration window. It features a title bar with a folder icon and the text 'New Document Type'. The main area contains several input fields:

- Document Type:** 0001
- Description:** Sales Order
- ORDER_FULFILLMENT:**
 - Process Type:** ORDER_FULFILLMENT
 - Process Type Name:** Order Fulfillment
 - Description:** Order Fulfillment
- ORDER_NEGOTIATION:**
 - Process Type:** ORDER_NEGOTIATION
 - Process Type Name:** Order Negotiation
 - Description:** Order Negotiation
- ORDER_DELIVERY:**
 - Process Type:** ORDER_DELIVERY
 - Process Type Name:** Outbound Shipment
 - Description:** Outbound Shipment

3. In Document Type, enter the new document type identification number.
4. In Description, enter a brief description of the document type.
5. For each process type associated with the document type you are saving as, enter the process type, the process type name, and a brief description.
6. Choose . The new document type appears in the document type tree with the associated process types.

Note: An .ex extension is automatically appended to the document type and process type values you have specified.

4.1.3 Modifying a Document Type's Description

You can modify a document type's description.

To modify a document type's description:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Document Types swimlane, right-click on the applicable document type and choose Details. The Document Type Details window appears.



3. In Description, enter the new description.
4. Choose .

4.1.4 Modifying a Process Type

You can define the parameters and templates that are particular to an individual process type. These definitions are applied to a document throughout its lifecycle in the process type.

To modify a process type:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Details. The Process Type Details window appears.

The screenshot shows the 'Fulfillment Process Type Details' window. It has two tabs: 'Primary Info' and 'Templates'. The 'Primary Info' tab is active and contains the following fields and options:

- Process Type: ORDER_FULFILLMENT
- Process Type Name: Order Fulfillment
- Description: Order Fulfillment
- Represents A Return
- Use As A Template Order
- Validate Item During Order Creation/Modification
- Driver Date:
 - Requested Ship Date
 - Requested Delivery Date

The 'Templates' tab is also visible and contains the following options:

- Use Template Order For Defaulting
- Log Audits For Draft Order
- Template Document Type: Template Order (dropdown menu)
- Default Delivery Method Based On Catalog

Use the Process Type Details window for:

- [Defining a Process Type's Primary Information](#)
- [Defining a Process Type's Templates \(Fulfillment Process Types Only\)](#)

4.1.4.1 Defining a Process Type's Primary Information

You can define a process type's parameters for order creation, inventory, financial transactions, and other related entities. These parameters are applied to a document throughout its lifecycle in the process type.

To define a process type's primary information:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Details. The Process Type Details window appears.
3. Choose the Primary Info tab.

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

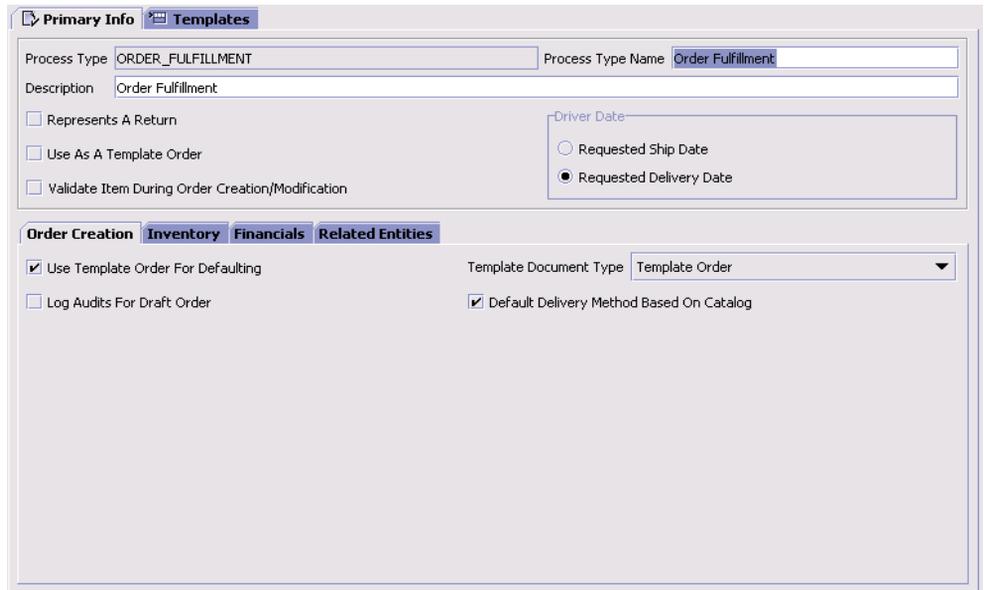


Table 4–1 Process Type’s Primary Info Tab

Field	Description
Process Type	The process type ID.
Process Type Name	Enter the name of the process type.
Description	Enter a brief description of the process type.

Table 4–1 Process Type's Primary Info Tab

Field	Description
Represents a Return	<p>Select this field if the process type is for return documents.</p> <p>When the Represents A Return is selected:</p> <ul style="list-style-type: none"> • The Update Supply For Buyer Organization field cannot be selected. • The Create Reservation On Order Creation field cannot be selected. <p>Selecting Represents a Return has the following impact throughout the system:</p> <ul style="list-style-type: none"> • At the line level, the allocation date is ignored and the Schedule and Release setting is always selected. • The payment status is set to AUTHORIZED instead of AWAITING AUTHORIZATION when creating an order. • All invoices created for orders of this document type have an Invoice Type of RETURN. • The document type order's charge transaction records have a charge type of RETURN. • The Schedule time-triggered transaction does not check for inventory and always creates a release.
Use as a Template Order	Select this field if documents in this process type can be used as a template for another document.
Validate Item During Order Creation/ Modification	Select this field if you want item IDs and unit's of measure to be validated against Yantra 7x Product Management application (or external product management application) upon order creation or adding an order line.
Driver Date	
Requested Ship Date	Select this option if you want the fulfillment process to be driven by the order document's requested ship date.
Requested Delivery Date	Select this option if you want the fulfillment process to be driven by the order document's requested delivery date.

Table 4–1 Process Type's Primary Info Tab

Field	Description
Order Creation Tab	
Use Template Order For Defaulting	<p>Select this field if you want to be able to default some attributes of an existing order into a newly created order.</p> <p>When an order is created and this field is selected, the system looks for an existing order with the same Buyer, Seller, and Enterprise organizations, as well as the document type specified in the Template Document Type field. If an existing template is found, some of the attributes may be copied into the new order.</p> <p>For more information about template orders, see the <i>Yantra 7x Distributed Order Management User Guide</i>, <i>Yantra 7x Supply Collaboration User Guide</i>, and/or <i>Yantra 7x Reverse Logistics User Guide</i>.</p>
Template Document Type	If you choose Use Template Order For Defaulting, select the document type you want to use for the default template order.
Log Audits For Draft Order	Select this field if you want the system to log audit records when modifications are made to orders in draft status.
Refund Fulfillment Order Document Type	Select the document type you want to use from refund fulfillment orders.
Default Delivery Method Based On Catalog	<p>Select Default Delivery Method Based On Catalog if you want the system to default to the configured delivery method assigned to items set up as Delivery Allowed in the catalog. For more information about defining item attributes, see the <i>Yantra 7x Product Management Configuration Guide</i>.</p> <p>Note: This field should not be selected for any document types other than Sales Orders.</p>
Inventory Tab	

Table 4–1 Process Type's Primary Info Tab

Field	Description
Demand Type For Schedule	<p>Select the demand type to be considered when the Schedule Order time-triggered transaction is run.</p> <p>Important: All the supply types that you have associated with the demand type in the Inventory Considerations table are considered for scheduling inventory to satisfy demand optimization. For more information about configuring inventory considerations, see the <i>Yantra 7x Inventory Synchronization Configuration Guide</i>.</p> <p>Note: The demand type should have the same supply types associated with it as the demand type of the transaction that you plan for Schedule Order to drop into for this process type.</p>
Demand Type for Release	<p>Select the demand type to be considered when the Release Order time-triggered transaction is run.</p> <p>Important: All the supply types that you have associated with the demand type in the Inventory Considerations table are considered for release inventory calculations. For more information about configuring inventory considerations, see the <i>Yantra 7x Inventory Synchronization Configuration Guide</i>.</p>
Allow Inventory Updates For The Seller Organization	Select this field if you want the system to perform inventory updates for the Seller.
Allow Inventory Updates For The Buyer Organization	Select this field if you want the system to perform inventory updates for the Buyer.
Allow Inventory Check During Schedule And Release	Select this field if you want inventory supply and demand data stored in the system to be used for availability calculation during the schedule and release processes.
Create Reservation On Order Creation	Select this field if you want to enable reservations during order creation.
Financials Tab	
Allow Invoice Creation	Select this field to enable invoice creation for orders or shipments.

Table 4–1 Process Type's Primary Info Tab

Field	Description
Allow Pro Forma Invoice Creation For Shipments	<p>Select this field to enable Pro Forma invoice creation for shipments. For more information on Pro Forma invoicing, refer to <i>Yantra 7x Product Concepts</i>.</p> <p>Note: This flag is only applicable to shipments created from orders. This flag is ignored for shipments without orders. For more information on creating Pro Forma invoices for shipments without orders, refer to the <i>Yantra 7x Javadocs</i>.</p>
Allow Payment Processing	<p>Select this field to enable payment processing. For more information on payment processing, refer to <i>Yantra 7x Product Concepts</i>.</p>
Allow Price Calculation For Draft Orders	<p>Select this field if you want to be able to associate a price program with an order during draft order creation. For more information about price programs, see the <i>Yantra 7x Distributed Order Management Configuration Guide</i>.</p> <p>Note: If both this field and Allow Price Calculation For Confirmed Orders are not selected, no price calculations are performed on orders, even if they have price programs associated with them. If both this field and Allow Price Calculation For Confirmed Orders are selected, price calculations can be performed at any point in an order's creation cycle. If only this field is selected, price calculations are only done at the time of draft order creation.</p>
Allow Price Calculation For Confirmed Orders	<p>Select this field if you want pricing to be done upon draft order confirmation and order creation.</p> <p>Note: If both this field and Allow Price Calculation For Draft Orders are not selected, no price calculations are performed on orders, even if they have price programs associated with them. If both this field and Allow Price Calculation For Draft Orders are selected, price calculations can be performed at any point in an order's creation cycle. If only this field is selected, price calculations are only done at any point after orders have been confirmed where it is applicable to do so (for example, after a quantity adjustment).</p>
Related Entities	
Allow Chained Order Creation	<p>Select this field if you want the order document to have the ability to create chained orders during scheduling.</p>

Table 4–1 Process Type's Primary Info Tab

Field	Description
Chained Drop Ship Order Document Type	Select the document type you want to use for the chained order document in a drop ship scenario. The drop ship scenario occurs when the final shipping point is the customer and the order is scheduled to a node external from your organization.
Consolidate New Order Releases	Select this field if you want the system to attempt to consolidate new releases into existing releases that have not been processed.
Chained Procurement Purchase Order Document Type	Select the document type you want to use for the chained order document in a procurement purchase order scenario. The procurement purchase order scenario occurs when the final shipping point to the customer is one of your nodes and the shipping node does not have enough stock and needs to be replenished from an external organization's node.
Allow Propagation Of Changes To Derived Parent	Select this field if you want changes on a derived order to be propagated to its derived parent when the appropriate listener transaction is configured. Note: If this field is not selected, a derived order's quantity can be greater than the parent order that is was derived from.
Chained Procurement Transfer Order Document Type	Select the document type you want to use for the chained order document in a procurement transfer order scenario. The procurement transfer order scenario occurs when the final shipping point to the customer is one of your nodes and the shipping node does not have enough stock and needs to be replenished from another node within your organization.
Create Shipments For Products Being Delivered In Addition To Work Order	If this is checked, shipments will be created for products that are being delivered in addition to the work order.

4.1.4.2 Defining a Process Type's Templates (Fulfillment Process Types Only)

Document templates are used at various times throughout Yantra 7x. The template type indicates how it is used. Typically, templates are required in scenarios in which a particular set of attributes of a given entity need to be considered for processing. For example, when calling the `copyOrder()` API, the Copy Order template is used to indicate which order attributes should be copied.

You can determine which XML attributes and elements should be included or excluded from master template XMLs for a given fulfillment process type.

To define a process type's XML templates:

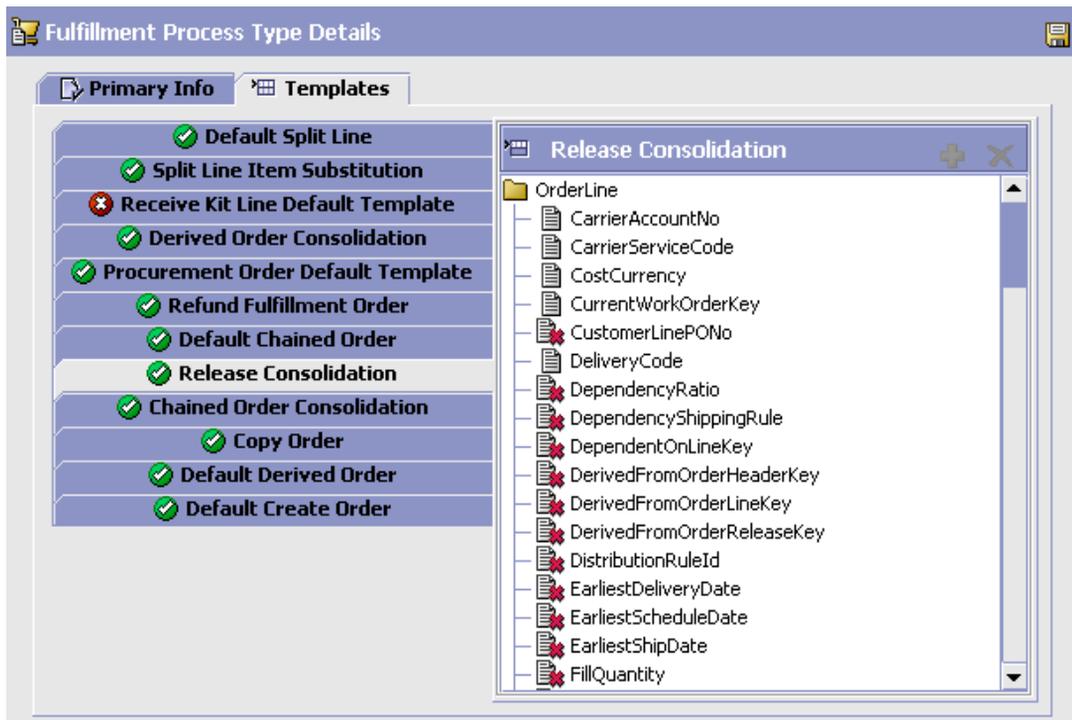
1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Details. The Process Type Details window appears.
3. Choose the Templates tab. The available master templates for the fulfillment process type you are working with are displayed as tabs. These master templates are retrieved from the `YFS_BASE_DOCUMENT_TYPE` table.
4. Choose the tab of the applicable master template. The master template XML is loaded from the `YFS_BASE_DOCUMENT_TEMPLATE` table and is combined with the template XML stored for this document type in the `YFS_DOCUMENT_TEMPLATE` table. Extended attributes are also added to the master template for each element that has extended attributes. The resulting XML is shown in hierarchical format in the tree.

Note: The extended branch of the template XML is automatically generated. It is not stored in either the `YFS_DOCUMENT_TEMPLATE` or `YFS_BASE_DOCUMENT_TEMPLATE` tables.

5. Choose  to include an XML attribute or element in the master template XML for this process type. Choose  to exclude an XML attribute in the master template XML for this process type.

Important: If you want to exclude all of an element's attributes, you must exclude the entire element.

Note: Some attributes are mandatory and cannot be excluded from the template.



4.2 Defining Process Type Pipelines

You can define your business process workflow by creating process type pipelines. 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.

When you choose Model Process from the Process Modeling tree, two frames are displayed. The left-hand frame is the repository and the right-hand frame is the work area.

There are six available tabs at the bottom of the repository. These tabs allow you to switch between the Pipeline tree, Transaction tree, Status tree, Condition tree, Action tree, and Service tree. When you choose an entity from any of these trees, its details are displayed in the work area frame, if applicable.

When configuring a pipeline, you can enable auto-hints to aid you in your configuration process. To activate auto-hints, right-click anywhere in the work area and choose Enable Auto Hint. When auto-hints are activated, transactions that a particular drop status can connect to are highlighted in a blue frame when you are setting up a graphical pipeline.

You can use Process Model for:

- [Defining Pipeline Determination](#)
- [Creating a Pipeline](#)
- [Saving a Pipeline as a Draft](#)

- [Modifying a Pipeline](#)
- [Defining a Pipeline's Monitoring Rules](#)
- [Defining Transactions](#)
- [Defining Statuses](#)
- [Defining Conditions](#)
- [Defining Actions](#)

4.2.1 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.

4.2.1.1 Condition Variables for Pipeline Determination

For a list of the condition variables that can be used for pipeline determination, refer to [Appendix F, "Condition Builder Attributes"](#) on page 661.

4.2.2 Creating a Pipeline

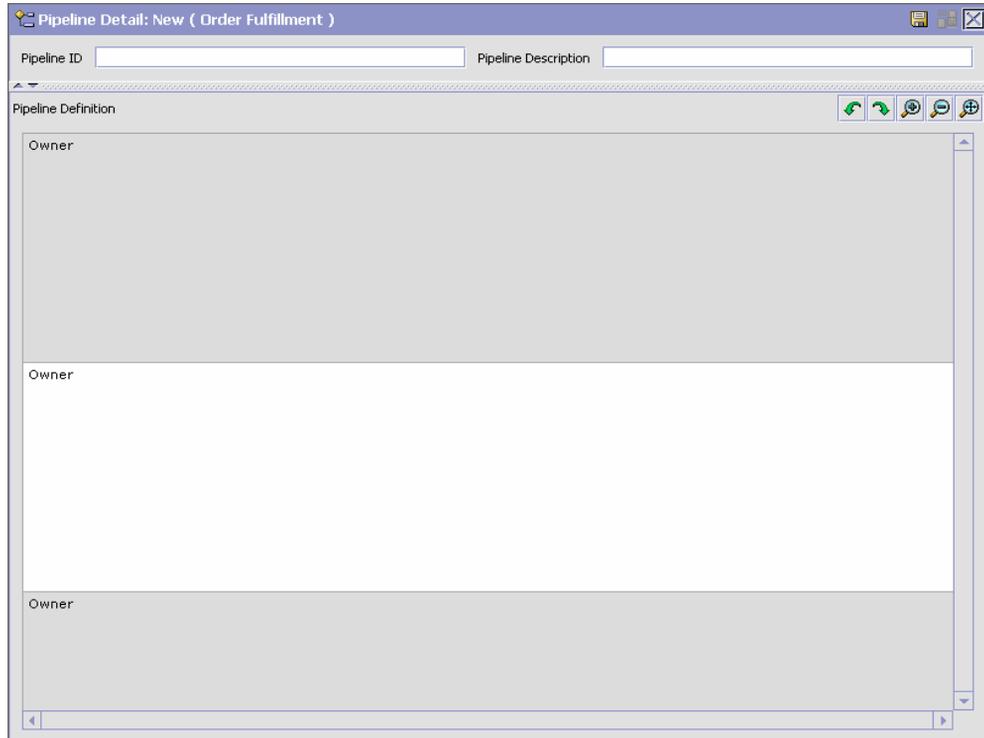
You can create a pipeline for the process type that you are working with.

To construct a pipeline you can drag transactions and conditions into the pipeline work area. Each transaction has a set of branches relating to each drop status. To link a transaction to another transaction you must drag the appropriate port from the first transaction to the second. You can identify what status belongs to which port by putting the arrow over the transaction's ports. You can link transactions back to themselves assuming they are allowed to pick up the status being linked back to.

Transactions can also be linked to conditions. To specify that you are extending the drop status with a condition, drag the port to the applicable condition and then to the pickup transaction. If the pickup status has the same base as the port, the link is allowed. Once the link is made, it is defaulted to the first possible pickup status.

To create a pipeline:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Pipeline tab.
4. Select Pipelines and choose . The Pipeline Detail window is displayed in the work area.



5. Drag and drop the applicable transactions and conditions into the work area and connect them following the rules detailed in this section.
6. Choose .

4.2.2.1 Condition Variables for Pipeline Setup

For a list of the condition variables that can be used for pipeline setup, refer to [Appendix F, "Condition Builder Attributes"](#) on page 661.

4.2.3 Saving a Pipeline as a Draft

You can save an incomplete pipeline as a draft. This draft can be retrieved for a final save without any necessary validations. When you save and activate the pipeline, the draft pipeline is deleted from the system.

To save a service as a draft:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Pipeline Tab.
4. Configure a pipeline as per the rules detailed in this section.
5. Choose . The pipeline is saved as a draft service.
6. When you are ready to save it as a complete and functional pipeline, choose .

Note: During runtime, when a status is reached that has been configured to be picked up by more than one TaskQ-based transaction in the pipeline, Yantra will create a TaskQ record for each of those transactions.

This includes situations where the Yes and No branches of a condition both drop into the same status, but feed into different TaskQ-based transactions.

Note: When you save a pipeline as a draft, any existing draft for the pipeline is overwritten. When you save the draft as an actual pipeline, any existing pipeline with the same pipeline ID is overwritten.

4.2.4 Modifying a Pipeline

Note: Yantra 7x does not support modification of pipelines while it is processing transactions. Modify pipelines only on a quiet system, when no APIs or agents are running.

If high availability is a concern, never modify pipelines directly in production. Make all changes in your test environment then migrate the changes to your production environment using the Yantra 7x Configuration Deployment Tool.

To modify a pipeline:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Pipeline tab.
4. Expand the Pipelines branch.
5. Select the applicable pipeline and choose . The Pipeline Detail window is displayed in the work area.
6. Using the rules and concepts detailed in the Basics section of this chapter, modify the pipeline according to your business practices. For more information about how use the drag and drop window, see [Section 2.2.2.4, "Drag and Drop Window"](#) on page 22.
7. Choose .

Note: During runtime, when a status is reached that has been configured to be picked up by more than one TaskQ-based transaction in the pipeline, Yantra will create a TaskQ record for each of those transactions.

This includes situations where the Yes and No branches of a condition both drop into the same status, but feed into different TaskQ-based transactions.

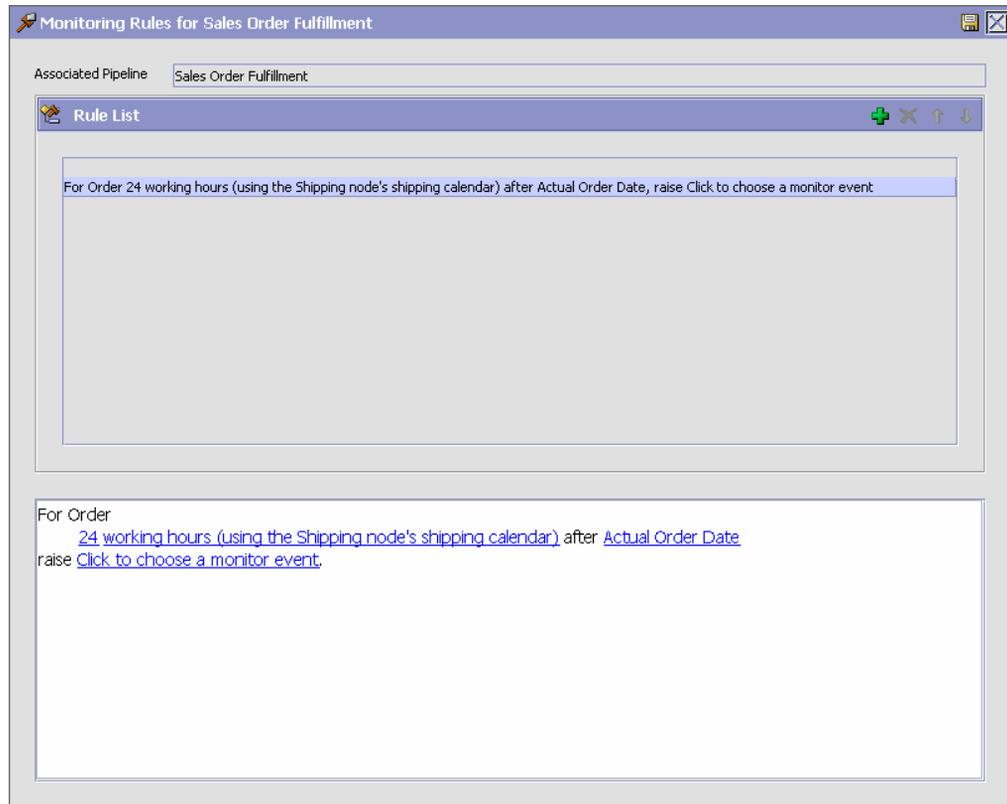
4.2.5 Defining a Pipeline's Monitoring Rules

Using the monitoring rule components you configured while defining the process type, you can define the parameters used to monitor orders and shipments throughout their lifecycle in fulfillment and shipment process type pipelines. For more information about configuring monitoring rule components, see the *Yantra 7x Distributed Order Management Configuration Guide*.

Note: You can only define a pipeline's monitoring rules if your organization owns the pipeline you are configuring them for.

To define a pipeline's monitoring rules:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Pipeline tab.
4. Expand the Pipelines branch.
5. Select the applicable pipeline and choose . The Monitor Rules window is displayed in the work area.



6. From Rule List, choose . The Rule Type pop-up window is displayed.



7. Select the rule type you was to define and add to the rule list. The rule details appears in the lower frame. You can select the hyper-text in the rule details and define the individual parameters. Refer to [Table 4–2](#) for a list of rule types and their configurable rule details.

Note: When defining the hours parameter, you can select whether the hours are based on elapsed hours or on the working hours for any calendars you may have defined.

8. Choose .

Note: You can increase or decrease a monitoring rule's priority by selecting the rule and choosing the up-arrow to increase it's priority and the down-arrow to decrease its priority.

Table 4–2 Monitoring Rule Types

Rule Type	Rule Details (Bold and Italicized Text Indicates Configurable Parameters)
Milestone has not reached before a date	If Order/Shipment has not reached a milestone <i>n calendar hours</i> before a date type , then raise a monitor event .
Milestone has not reached after a date	If Order/Shipment has not reached a milestone within <i>n calendar hours</i> of a date type , then raise a monitor event .
Milestone has not reached after another Milestone	If Order/Shipment has not reached a milestone within <i>n calendar hours</i> of a milestone , then raise a monitor event .
Milestone has reached before a date	If Order/Shipment has reached a milestone <i>n calendar hours</i> before a date type , then raise a monitor event .
Milestone has reached after a date	If Order/Shipment has reached a milestone within <i>n calendar hours</i> of a date type , then raise a monitor event .
Milestone has reached after another Milestone	If Order/Shipment has reached a milestone within <i>n calendar hours</i> of a milestone , then raise a monitor event .
Has been in a status	If Order/Shipment has been in a status for <i>n calendar hours</i> , then raise a monitor event .
Before a date	For Order/Shipment <i>n calendar hours</i> before a date type , raise a monitor event .
After a date	For Order/Shipment <i>n calendar hours</i> after a date type , raise a monitor event .
After a Milestone	For Order/Shipment <i>n calendar hours</i> after a milestone , raise a monitor event .
Date before another date	For Order/Shipment, if a date type is more than <i>n calendar hours</i> before a date type , then raise a monitor event .

Table 4–2 Monitoring Rule Types

Rule Type	Rule Details (Bold and Italicized Text Indicates Configurable Parameters)
Date within a specified time after another date	For Order/Shipment, if a date type is within <i>n calendar hours</i> after a date type , then raise a monitor event .
Date not within a specified time after another date	For Order/Shipment, if a date type is not within <i>n calendar hours</i> after a date type , then raise a monitor event .
Date after another date	For Order/Shipment, if a date type is more than <i>n calendar hours</i> after a date type , then raise a monitor event .
Conditionally Milestone has not reached before a date	If Order/Shipment meets a condition and has not reached a milestone <i>n calendar hours</i> before a date type , then raise a monitor event .
Conditionally Milestone has not reached after a date	If Order/Shipment meets a condition and has not reached a milestone within <i>n calendar hours</i> of a date type , then raise a monitor event .
Conditionally Milestone has not reached after another Milestone	If Order/Shipment meets a condition and has not reached a milestone within <i>n calendar hours</i> of a milestone , then raise a monitor event .
Conditionally Milestone has reached before a date	If Order/Shipment meets a condition and has reached a milestone <i>n calendar hours</i> before a date type , then raise a monitor event .
Conditionally Milestone has reached after a date	If Order/Shipment meets a condition and has reached a milestone within <i>n calendar hours</i> of a date type , then raise a monitor event .
Conditionally Milestone has reached after another Milestone	If Order/Shipment meets a condition and has reached a milestone within <i>n calendar hours</i> of a milestone , then raise a monitor event .
Conditionally has been in a status	If Order/Shipment meets a condition and has been in a status for <i>n calendar hours</i> , then raise a monitor event .
Conditionally before a date	If Order/Shipment meets a condition , raise a monitor event <i>n calendar hours</i> before a date type .
Conditionally after a date	If Order/Shipment meets a condition , raise a monitor event <i>n calendar hours</i> after a date type .
Conditionally after a Milestone	If Order/Shipment meets a condition , raise a monitor event <i>n calendar hours</i> after a milestone .

Table 4–2 Monitoring Rule Types

Rule Type	Rule Details (Bold and Italicized Text Indicates Configurable Parameters)
Conditionally date before another date	If Order/Shipment meets a condition and if a date type is more than <i>n calendar hours</i> before a date type , then raise a monitor event .
Conditionally date after another date	If Order/Shipment meets a condition and if a date type is more than <i>n calendar hours</i> after a date type , then raise a monitor event .
Conditionally date within a specified time after another date	For Order/Shipment meets a condition and if a date type is within <i>n calendar hours</i> after a date type , then raise a monitor event .
Conditionally date not within a specified time after another date	For Order/Shipment meets a condition and if a date type is not within <i>n calendar hours</i> after a date type , then raise a monitor event .

Monitoring Rule Configuration Example

According to your business practices, you need to monitor orders that have not been released 24 hours before their requested shipment date and raise alerts for such orders. Assuming that you have configured the necessary monitor rule components, including milestones, date types, and alert consolidation rules, for the process type you are working with, you can create a monitoring rule identifying these parameters.

Using the "Milestone has not reached before a date" monitoring rule type, you can configure the system to monitor when the Released milestone has not been reached 24 hours before an order's requested shipment date.

4.2.6 Defining 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.

In Yantra 7x, APIs are used to execute transactions. When an API is invoked, the transaction ID is determined based on the context the API was executed. The transaction ID identifies the transaction to be executed. Depending on the situation the transaction ID can be passed as an input parameter or it can be predefined for the invoking API. For more information about APIs, refer to the *Yantra 7x Javadocs*.

Some extended transactions that are created may require custom coding to implement logic for the transaction. However, you can derive new transactions from abstract transactions provided by Yantra 7x. A transaction derived from an abstract transaction contains specific details such as, statuses and triggering mechanisms that do not require custom coding. For example, if you are configuring an order document pipeline that requires several different types of order status change transactions, you can derive multiple extended transactions from the Change Order Status abstract transaction and configure them in your pipeline without requiring custom coding.

Transactions can be classified as one or more of the following types:

- Externally-triggered
- User-triggered
- Time-triggered

Externally-Triggered Transactions

An externally-triggered transaction is performed through Yantra 7x Service Definition Framework which calls a corresponding API within Yantra 7x to execute the transaction.

User-Triggered Transactions

A user-triggered transaction is executed manually through the Application Consoles, configured alert queue, or e-mail service.

Time-Triggered Transactions

A time-triggered transaction is executed on scheduled intervals. In Yantra 7x, a time-triggered transaction is also called an agent.

You can use the Transactions tab for:

- [Creating an Extended Transaction](#)

- [Creating an Extended Transaction that is Derived from an Abstract Transaction](#)
- [Modifying a Transaction](#)
- [Deleting a Transaction](#)

4.2.6.1 Creating an Extended Transaction

You can create new custom transactions in the process type you are working with. These transactions can then be used in pipeline creation and modification.

Important: When you are "creating" an extended transaction that is not derived from an abstract transaction, you are creating a custom transaction for which software development must be done separately before the extended transaction can be used within a pipeline.

To create an extended transaction:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Transactions tab.
4. Select the Transactions branch and choose . The Create New Transaction pop-up window is displayed.

Create New Transaction

When creating custom transaction, you can either create it as a derived transaction or a non-derived transaction. Yantra provides abstract transactions from which you may derive specific transactions for your scenario.

Choose an option:

Do not derive from an abstract transaction

Derive from this abstract transaction

OK Cancel

5. Select 'Do not derive from an abstract transaction' to create a new transaction and associate your own logic with it.
6. Choose OK. The Transaction Detail window is displayed in the work area.
7. Enter information in the applicable fields. Refer to [Table 4–3](#) for field value descriptions.
8. Choose .

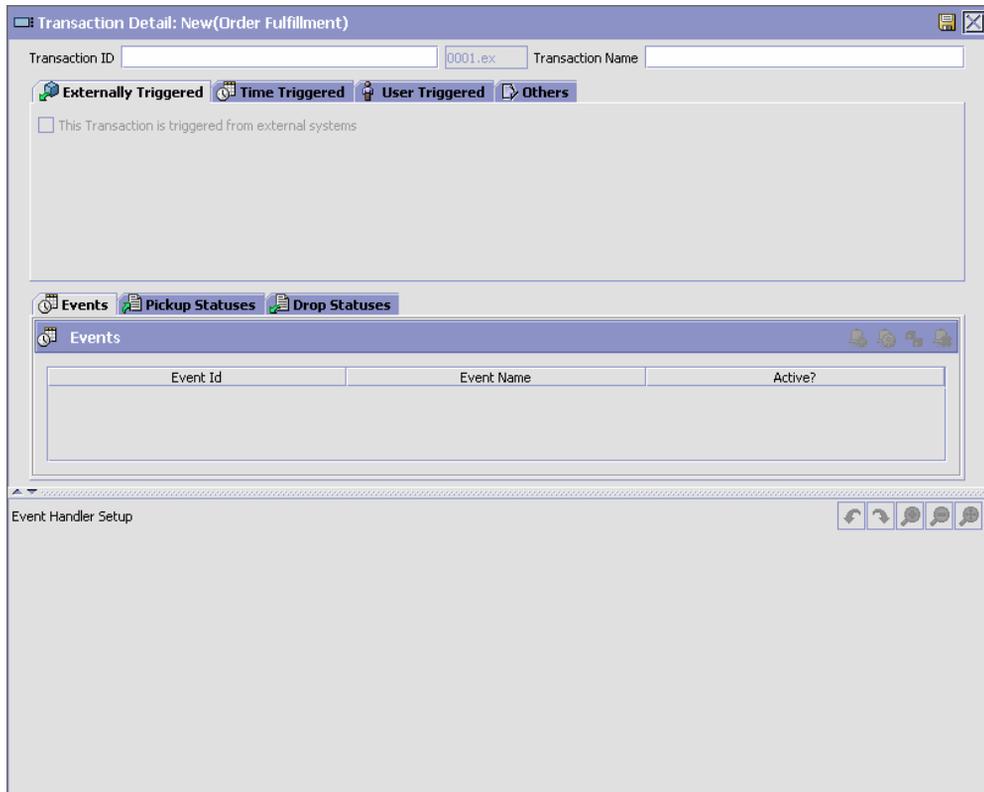


Table 4–3 Transaction Details Window

Field	Description
Transaction ID	Enter the transaction ID. Note: The document type and '.ex' are automatically appended to the transaction ID of transactions you create.
Transaction Name	Enter the transaction's name.
Externally Triggered	The Externally Triggered tab provides an interface to set up an externally-triggered transaction. For more information about setting up an externally-triggered transaction, see Section 4.2.6.1.1, "Specifying a Transaction as Externally-Triggered" on page 164.

Table 4–3 Transaction Details Window

Field	Description
Time Triggered	The Time Triggered tab provides an interface to set up a time-triggered transaction. For more information about setting up a time-triggered transaction, see Section 4.2.6.1.2, "Specifying a Transaction as Time-Triggered" on page 166.
User Triggered	The User Triggered tab provides an interface to set up a user-triggered transaction. For more information about setting up a user-triggered transaction, see Section 4.2.6.1.3, "Specifying a Transaction as User-Triggered" on page 174.
Other	
This transaction is task based	Select this field if your transaction is triggered by a task queue. This indicates that whenever the previous transaction in the pipeline completes its functions, the system creates a task in the task queue table for this transaction. Important: If this field is selected and the transaction is not identified as a time-triggered transaction, task queue entries are not created.
Works Based On	Select the order level for which the transaction is used (for example, order or order release).
Spawns another process?	Select this field if the transaction spawns another system process.
Spawning process type	If you chose 'Spawns another process?', select the process type the transaction spawns.
Chained Document Type	If a chained order is created from this transaction, enter the document type of the chained order that is created. For example, Sales Order or Purchase Order.
Derived Document Type	If a derived order is created from this transaction, enter the document type of the derived order that is created. For example, Return Order or Exchange Order.
This Transaction Is Derived From Abstract	Indicates if the transaction is derived from a base transaction.
Base Transaction Name	If the transaction is derived from a base transaction, the name of the base transaction is displayed.

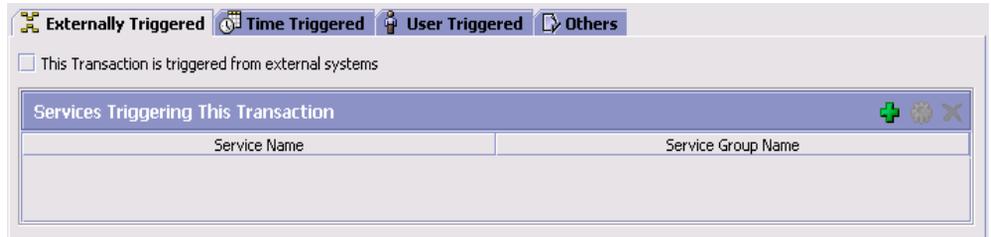
Table 4–3 Transaction Details Window

Field	Description
This Transaction Can Be Stopped From Processing Orders That Are On Hold	<p>Check this if you want this transaction to be hold type enabled. A hold type enabled transaction can be configured to be stopped from processing orders that are on placed on a particular hold type.</p> <p>This flag cannot be modified for system defined transactions, but can be set for all custom transactions that are not derived from an abstract transaction.</p>
Events	<p>An event is a specific occurrence in the business process, often creating a status change or generated alert. When an event occurs in a transaction an action is triggered.</p> <p>The Events tab provides an interface to set up events and event handlers. For more information about events and event handlers, see Section 4.2.6.1.4, "Adding an Event to a Transaction" on page 175, Section 4.2.6.1.7, "Defining Event Handlers" on page 177.</p>
Pickup Statuses	<p>The Pickup Statuses tab provides an interface to set up pickup statuses. For more information about pickup statuses, see Section 4.2.6.1.8, "Adding a Pickup Status to an Extended Transaction" on page 178.</p>
Drop Statuses	<p>The Drop Statuses tab provides an interface to set up drop statuses. For more information about pickup statuses, see Section 4.2.6.1.10, "Adding a Drop Status to an Extended Transaction" on page 180.</p>

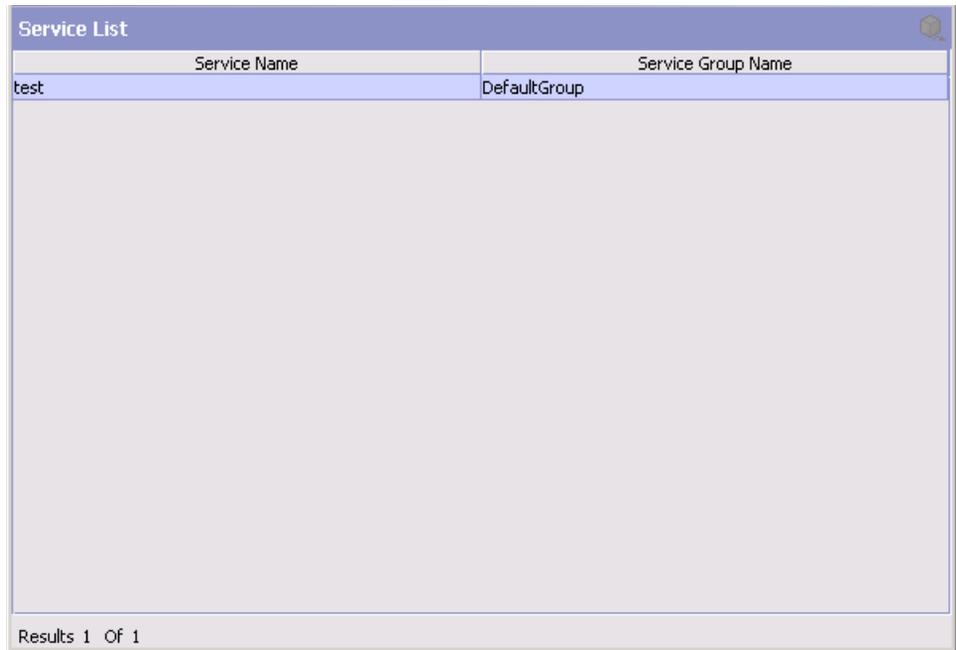
4.2.6.1.1 Specifying a Transaction as Externally-Triggered

To create an externally-triggered extended transaction:

1. From the Transaction Details window, choose the Externally Triggered tab.



2. Check 'This transaction is triggered from external systems' to indicate that this is an externally-triggered transaction.
3. From the Services Triggering This Transaction list, choose . The Service List pop-up window is displayed.



4. Select the services that trigger the transaction and choose . The services are added to the Services Triggering This Transaction list.
5. Continue to enter information in the applicable transaction fields. Refer to [Table 4-3](#) for field level descriptions.

- Choose .

4.2.6.1.2 Specifying a Transaction as Time-Triggered

You can set up a transaction to be triggered by Yantra 7x agents. For detailed information about time-triggered transactions, see [Appendix A, "Time-Triggered Transaction Reference"](#).

Note: If you are creating a time-triggered transaction for a derived transaction, please observe that agent criteria data is not automatically populated and must be created.

To create a time-triggered extended transaction:

- From the Transaction Details window, choose the Time Triggered tab.



- Check 'This transaction is time triggered' to indicate that this is a time-triggered transaction.
- In Java Class, enter the agent class that you want to handle agent messages.
- From the Agent Criteria table, choose . The Agent Criteria pop-up window displays.
- Enter information in the applicable fields. Refer to [Table 4-4](#) for field level descriptions.

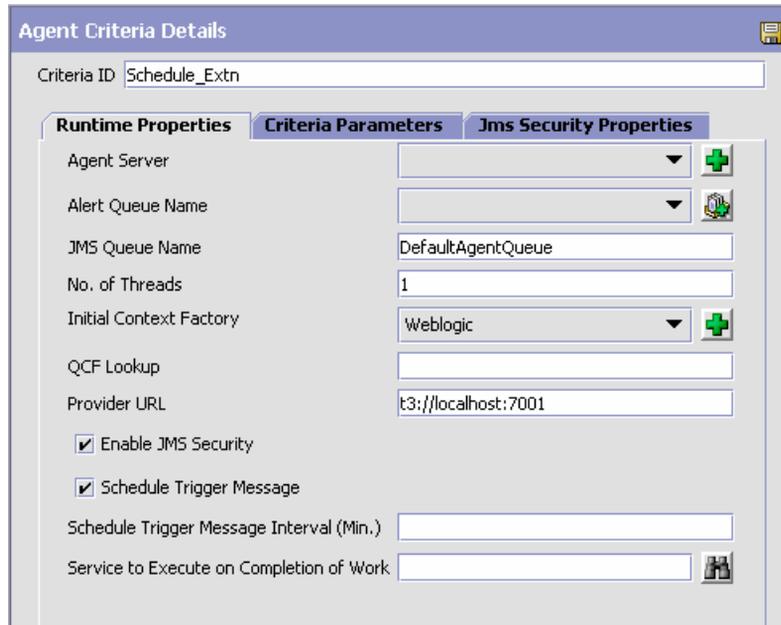


Table 4–4 Time-Triggered Transaction Runtime Properties

Control	Description
Runtime Properties	
Agent Server	The server on which this instance of the transaction is to be run. To add new Agent Servers, click the Add Servers button next to this field. This is a parameter used to start the agent server, for more information about this parameter, see the <i>Yantra 7x Installation Guide</i> .
Alert Queue Name	The name of the alert queue.
JMS Queue Name	The name of the JMS queue that contains messages to be processed by this transaction.
No. of Threads	The number of concurrent threads with which this transaction should be run.

Table 4–4 Time-Triggered Transaction Runtime Properties

Control	Description
Initial Context Factory	<p>The class providing an Initial Context implementation for your application server to enable remote Java clients to connect.</p> <p>Select WebSphere MQ if you are using MQSeries accessed through a WebSphere iiop URL. This sets the class name to: <code>com.ibm.websphere.naming.WsnInitialContextFactory</code>.</p> <p>Select File if you are using MQSeries accessed through a file URL, as with WebLogic. This sets the class name to <code>com.sun.jndi.fscontext.RefFSContextFactory</code>.</p> <p>Select WebLogic if you are using WebLogic JMS. This sets the class name to <code>weblogic.jndi.WLInitialContextFactory</code>.</p> <p>Note: You can override this value using the following <code>yfs.properties</code> parameter:</p> <pre>#yfs.agent.override.icf://<InitialContextFactory Name></pre> <p>Note: You can configure your own initial context factory settings to be used here. For more information about defining initial context factory codes, see the <i>Yantra 7x Platform Configuration Guide</i>.</p>
QCF Lookup	<p>The name of the queue connection factory. This name corresponds with a JMS connection factory configured in the application server cluster running Yantra 7x.</p> <p>Note: You can override this value using the following <code>yfs.properties</code> parameter:</p> <pre>#yfs.agent.override.qcf://<QueueConnectFactory Name></pre>

Table 4–4 Time-Triggered Transaction Runtime Properties

Control	Description
Provider URL	<p>The URL containing the protocol and address used to access the JMS queue.</p> <p>If you use WebLogic JMS, enter the following value:</p> <pre>t3://<IP address and port of the WLS instance></pre> <p>If you use MQSeries through a JNDI file, enter the following value:</p> <pre>file:/<pathname></pre> <p>If you use MQSeries through WebSphere JNDI, enter the following value:</p> <pre>iiop://<IP address and port of the WAS administration server></pre> <p>Note: You can override this value using the following yfs.properties parameter:</p> <pre>#yfs.agent.override.providerurl=t3://<ip Address>:<port></pre>
Enable JMS Security	<p>Check this box if you want JMS Security to be enabled. Once selected, the JMS Security Parameters tab is enabled to enter the name-value pairs.</p>
Schedule Trigger Message	<p>Check this box to configure the agent to execute the agent trigger periodically from within the Agent Server during runtime.</p> <p>When there are no messages for the agent to process, a new trigger message is sent to the agent at specified time intervals.</p>
Schedule Trigger Message Interval (Min)	<p>Enter the desired time interval in minutes.</p>
Service to Execute on Completion of Work	<p>Enter the service to be executed upon completing the execution of the selected agent. You can select the required service by using the  button.</p>
Criteria Parameters	
Parameter Name	<p>The name of parameter sent to the transaction. This is a parameter used to trigger the transaction, for more information about this parameter, see Appendix A, "Time-Triggered Transaction Reference".</p>

Table 4–4 Time-Triggered Transaction Runtime Properties

Control	Description
Parameter Value	The value of the parameter sent to the transaction. For valid names and parameters, see Appendix A, "Time-Triggered Transaction Reference" .
<p>JMS Security Properties Tab</p> <p>This is enabled upon selecting Enable JMS Security in the runtime properties tab. You can override the JMS security properties specified here by enabling the agent and flow authorization parameters in <code>yfs.properties</code>.</p> <p>For more information on application server-specific security mechanisms see "Setting up the JMS Security Properties" section.</p>	
Parameter Name	Enter the name of the security parameter.
Parameter Value	Enter the value of the security parameter.

6. Choose .
7. Continue to enter information in the applicable transaction fields. Refer to [Table 4–3](#) for field level descriptions.
8. Choose .
9. Restart the appropriate Agent Servers for the changes to take effect.

Setting up the JMS Security Properties

Based on your application server, you also need to pass the name-value pairs for user authentication.

For WebLogic specify the following name-value pairs in the parameter name and values explained in [Table 4–4](#):

- `java.naming.security.principal`=<user ID configured in the WebLogic server and assigned to the queue>
- `java.naming.security.credentials`=<password for the above user ID as configured for the WebLogic server>

Note: This is the standard way of setting up security on JNDI in the currently supported version of WebLogic. For more information on the authentication mechanism, setting up queues and QCF, refer to your WebLogic documentation.

For WebSphere and WebSphere MQ set up the desired forms of authentication and encryption where appropriate. Additionally, modify the java commands as described below to suit the desired goal.

Before you modify, make sure you have defined the following variables in your environment:

- WAS_HOME refer to the installation directory of the WebSphere software
 - MQ_HOME refers to the installation location of the WebSphere MQ software.
 - PROFILE_NAME refers to the name of the profile in which you created the server.
1. To allow agents to be authenticated to WebSphere JNDI, add the following definitions:
 - `-Djava.ext.dirs=<CLASSPATH>`, where the CLASSPATH should contain the following directories:
 - * `$MQ_HOME\java\lib`
 - * `$WAS_HOME\AppServer\java\jre\lib\ext`
 - * `$WAS_HOME\AppServer\java\jre\lib`
 - * `$WAS_HOME\AppServer\classes`
 - * `$WAS_HOME\AppServer\lib`
 - * `$WAS_HOME\AppServer\lib\ext`
 - * `$WAS_HOME\AppServer\properties`
 - * `$WAS_HOME\AppServer\profiles\<PROFILE_NAME>\properties.`
 - `com.ibm.CORBA.ConfigURL` should be set to the full path to the sas props file that you want to use such as

```
-Dcom.ibm.CORBA.ConfigURL=$WAS_  
HOME/AppServer/profiles/<PROFILE_  
NAME>/properties/sas.client.props.
```

The SAS props file is obtained from the WebSphere installation. You need to modify this text file to contain the username and password to be used for authentication to the WebSphere (corbaloc based) JNDI.

Note: For more information on how to set any of the above mentioned defines refer to IBM documentation. In specific, read the WebSphere documentation to understand how to enable and configure Global security.

2. To enable SSL encryption on the transmission of JMS messages to MQ, enable SSL on the channel to which your agents and services are connected. Create the QCF using the equivalent SSLCIPHERSPEC. On the java command line specify the following definitions:
 - javax.net.ssl.trustStore
 - javax.net.ssl.keyStorePassword
 - javax.net.ssl.KeyStore

Note: You need to refer to the WebSphere MQ documentation to learn how to turn on SSL on the server channel to which the Yantra 7x agents and services connect. For more information on how to use the SSLCIPHERSPEC option while creating the QCF see the IBM documentation.

Adding a New Server

You can add a new server from the Agent Criteria Details screen or from the Service Definition Framework. This screen provides the options for you to terminate the server on completing the task.

For example, in a once a day wave release scenario, the orders are downloaded through an integration server, waves are created and batched, pick locations are assigned, waves are released and printed in multiple servers. This may take an hour or more to process, but once

completed the servers are idle and waiting until the next day. Even though the processes are idle, they consume valuable resources like memory and CPU upon the server.

To avoid this idle time, you can configure the server to terminate automatically. To achieve this you can specify certain options in the Server Details upon creation as described in [Table 4–5, "Server Details"](#).



Table 4–5 Server Details

Field	Description
Server Properties Tab	
Server Name	Enter the name of the server.
Terminate Server on Idle	Select this option if you want to terminate the server when the task is completed or when idle. Once this option is selected the next two fields are enabled.
Startup Delay for Termination Monitor (minutes)	Enter the monitor start time. This is to ensure that the server does not terminate before it has completed one successful execution.
Termination Monitor Interval (minutes)	Enter the idle wait time before terminating the server.
Sub Service List Tab	
Subflow Name or Criteria ID	Lists the name of the subflow or the criteria belonging to the configured service or agent.
Threads	Specifies the number of threads.

Click  upon entering the details.

Once the server which has been configured to terminate is started, it monitors the threads to check if they are idle. The monitor start time indicates the time the number of minutes delay before it starts. Once all the threads are idle, the server waits the configured amount of time before terminating. If a new message comes in, the time is reset and the server again starts monitoring the threads.

4.2.6.1.3 Specifying a Transaction as User-Triggered

You can create a transaction that is triggered by the user.

To create a user-triggered extended transaction:

1. From the Transaction Details window, choose the User Triggered tab.



The screenshot shows a window titled 'Transaction Details' with four tabs: 'Externally Triggered', 'Time Triggered', 'User Triggered', and 'Others'. The 'User Triggered' tab is selected. Below the tabs, there is a checkbox labeled 'This transaction is triggered by Users' which is checked. Below the checkbox is a dropdown menu labeled 'When ready, notify user using' with a downward arrow.

2. Check 'This transaction is triggered by Users' to indicate that this is a user-triggered transaction.
3. From 'When ready, notify user using', select the service that should be triggered when a document enters this transaction's pick-up status. For example, if all orders created for a particular order type need to be verified by a customer service representative when an order is created, you can create a user-triggered transaction that invokes a service that sends an e-mail to the representative that verifies the order.

Important: You must select the applicable transaction order level from the Works Based On field on the Others tab for user notification to occur. For example, if you are configuring a transaction that is triggered by the user at the order release level, you must select Process Task Type for Order Release from the Works Based On drop-down.

4. Continue to enter information in the applicable transaction fields. Refer to [Table 4–3](#) for field value descriptions.
5. Choose .

4.2.6.1.4 Adding an Event to a Transaction

You can add events to transactions. These events signify occurrences in the process type's workflow and call associated actions.

To add an event to a transaction:

1. From the Transaction Detail window, choose the Events tab.



2. Choose . The Event Details pop-up window is displayed.
3. Enter information in the applicable fields. Refer to [Table 4–6](#) for field level descriptions.
4. Choose OK.

The screenshot shows a dialog box titled "Event Detail: New". It has a light blue header bar with a bell icon. Below the header, there are two text input fields: "Event ID" and "Event Name". Underneath these are three checkboxes: "Is Active?", "Can Enterprise Configure Event Handler?", and "Requires Backward Compatibility". To the right of the "Requires Backward Compatibility" checkbox is a "Version" dropdown menu. At the bottom right of the dialog are two buttons: "OK" and "Cancel".

Table 4–6 *Event Details Pop-Up Window*

Field	Description
Event ID	Enter the event ID.
Event Name	Enter the event’s name.
Is Active?	Select this field if the event is active for the transaction. Leave this field unselected to deactivate the event.
Can Enterprise Configure Event Handler?	If you are logged in as a Hub role, select this field if you want to allow Enterprise roles to be able to configure event handlers for the transaction.
Requires Backward Compatibility	Select this field if the event handler contains properties that require backward compatibility. If you select this field, choose the applicable version.

4.2.6.1.5 Modifying an Extended Transaction’s Event

To modify a transaction’s event:

1. From the Transaction Detail window, choose the Events tab.
2. Select the applicable event and choose . The Event Details pop-up window is displayed.

3. Modify information in the applicable fields. Refer to [Table 4–6](#) for field value descriptions.
4. Choose OK.

4.2.6.1.6 Deleting an Event from a Transaction

To delete an event from a transaction:

1. From the Transaction Detail window, choose the Events tab.
2. Select the applicable event and choose .

4.2.6.1.7 Defining Event Handlers

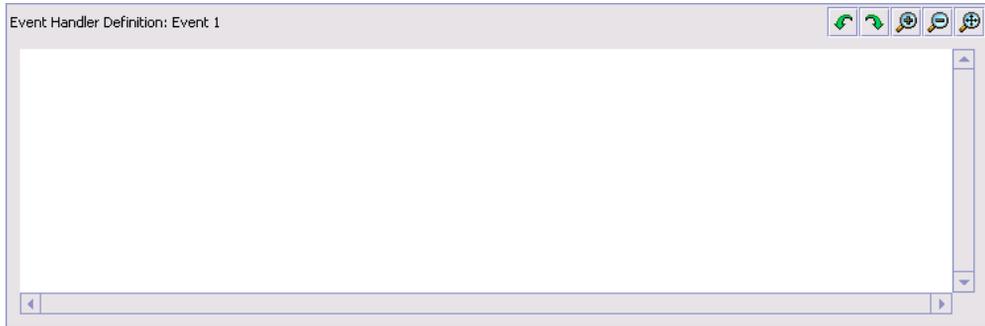
You can define event handlers that determine the types of actions that are performed when an event in a transaction occurs. You can provide conditions that apply to the event handler.

Important: Event handlers defined for a transaction in a particular pipeline are also applicable if the same transaction is used in another pipeline.

Note: When associating a condition with an event, refer to the *Yantra Tx Javadocs* to ensure that the applicable condition variables coincide with the event's key data and data published.

To set up event handlers:

1. From the Transaction Detail window, choose the Events tab.
2. Select the applicable event and choose the Configure Event Handler button. The Event Handler Definition work area activates.



3. Drag the applicable actions and conditions into the work area and connect them as per the rules detailed in this section.
4. Choose .

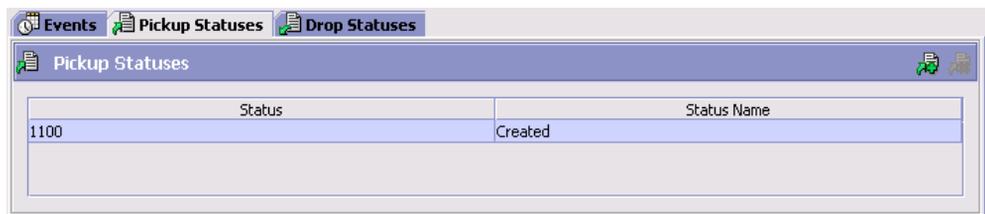
4.2.6.1.8 Adding a Pickup Status to an Extended Transaction

You can add a pickup statuses to extended transactions. A pickup status pulls the document from the preceding drop status and brings it into the transaction.

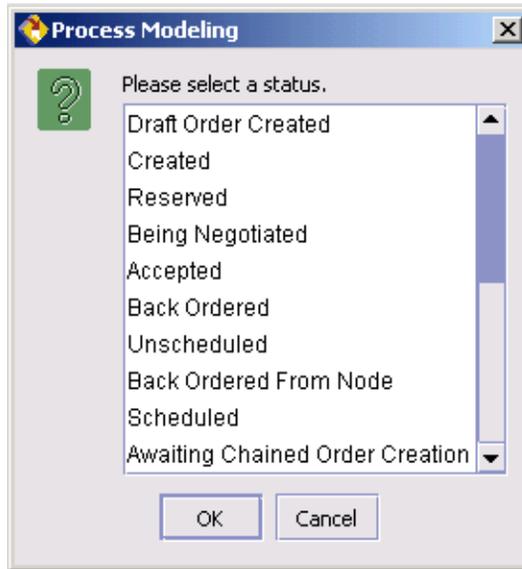
Important: While you cannot add pickup statuses or drop statuses to a system transaction, you can use an extended status as a pickup or drop status as long as its base status is included in the transaction's pickup or drop statuses.

To add a pickup status to a transaction:

1. From the Transaction Detail window, choose the Pickup Statuses tab.



2. Choose . The Select Status pop-up window is displayed.



Note: If the transaction has been derived from an abstract transaction, the pickup statuses populating the list are determined by the pickup status filter as defined in the derived transaction's base transaction.

3. Select the pickup status you want to add to the transaction.
4. Choose OK.

4.2.6.1.9 Deleting a Pickup Status from a Transaction

To delete a pickup status from a transaction:

1. From the Transaction Detail window, choose the Pickup Statuses tab.
2. Select the applicable pickup status and choose .

Note: You cannot delete an extended transaction's pickup status if it is the transaction's only pickup status and drop statuses exist for the transaction.

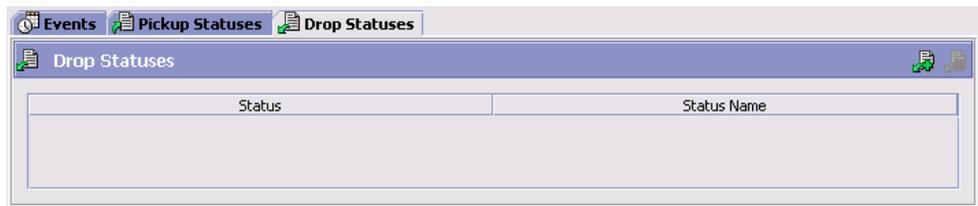
4.2.6.1.10 Adding a Drop Status to an Extended Transaction

You can add drop statuses to extended transactions. This status moves the document from the current transaction to the next transaction's pickup status.

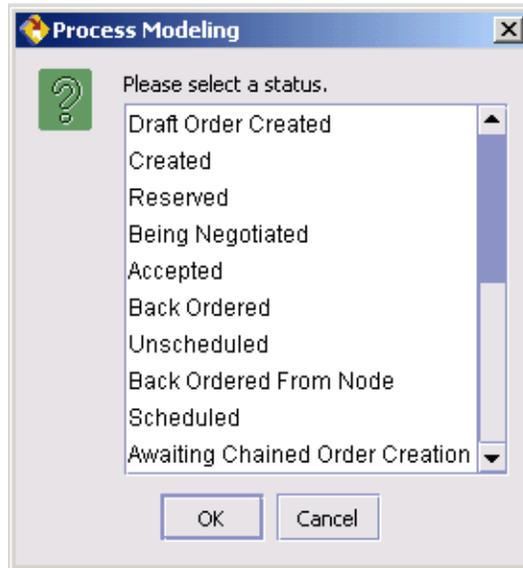
Important: While you cannot add pickup statuses or drop statuses to a system transaction, you can use an extended status as a pickup or drop status as long as its base status is included in the transaction's pickup or drop statuses.

To add a drop status to a transaction:

1. From the Transaction Detail window, choose the Drop Statuses tab.



2. Choose . The Select Status pop-up window is displayed.



Note: If the transaction has been derived from an abstract transaction, the drop statuses populating the list are determined by the drop status filter as defined in the derived transaction's base transaction.

3. Select the drop status you want to add to the transaction.
4. Choose OK.

4.2.6.1.11 Deleting a Drop Status from a Transaction

To delete a drop status from a transaction:

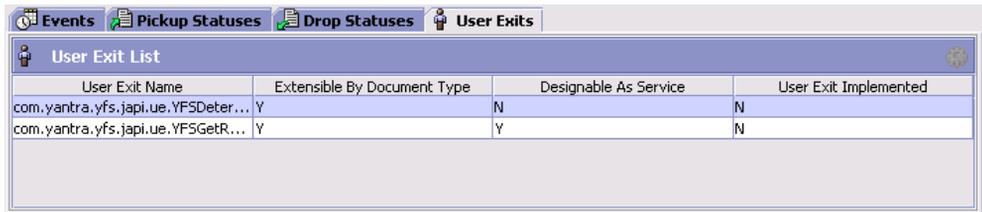
1. From the Transaction Detail window, choose the Drop Statuses tab.
2. Select the applicable drop status and choose .

4.2.6.1.12 Managing a Base Transaction's User Exits

User exits are Java interfaces which can be implemented for creating custom logic components. Once implemented, they must be configured so that Yantra 7x transactions can invoke them to perform the necessary logic at runtime.

To manage a base transaction's user exits:

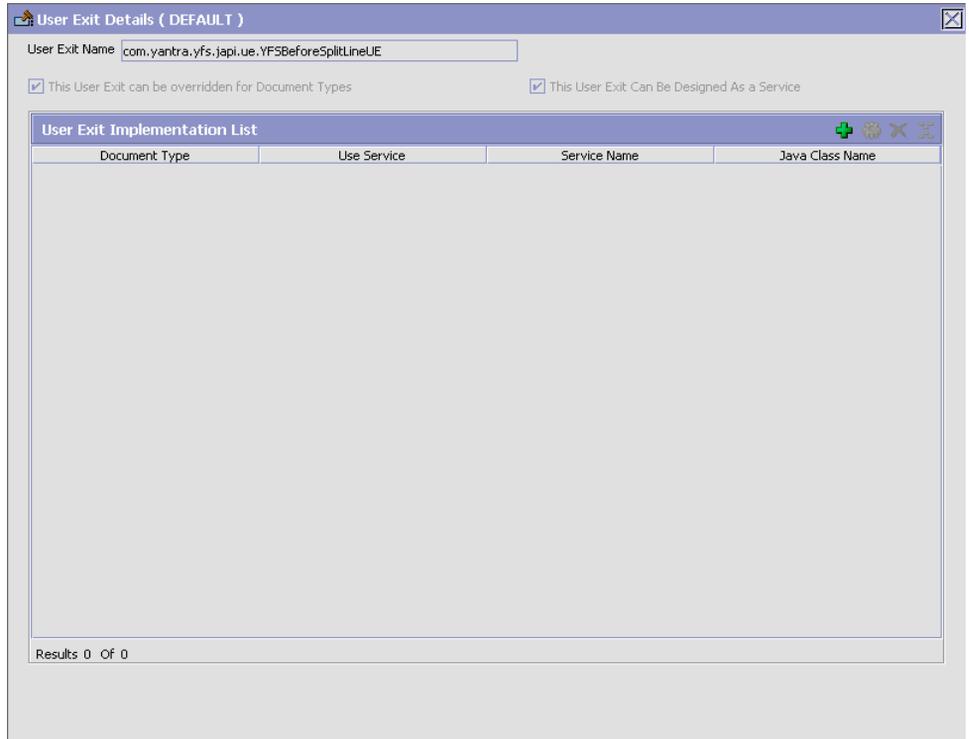
1. From the Transaction Detail window, choose the User Exits tab.



User Exit Name	Extensible By Document Type	Designable As Service	User Exit Implemented
com.yantra.yfs.japi.ue.YF5Deter...	Y	N	N
com.yantra.yfs.japi.ue.YF5GetR...	Y	Y	N

If the user exit can be implemented for a document type, the 'Can Override For Document Types' column displays 'Y'. If the user exit can be implemented for services, the 'Can Attach Service' column displays 'Y'. If the user exit is implemented, the 'User Exit Implemented' column displays 'Y'.

2. Locate the applicable user exit and choose . The User Exit Details window is displayed.



3. From the User Exit Implementation List table, choose . User Exit Implementation Details appears.
4. Enter information into the applicable fields. Refer to [Table 4–7](#) for field value descriptions.

User Exit Implementation Details

Document Type

Implement As a Service

Implement As a Java Class

Requires Backward Compatibility

Restrict Number Of Calls

Pool Size

Maximum Queue Length

Wait Time (seconds)

User Exit Implementation Notes

Table 4–7 User Exit Implementation Details Fields

Field	Description
User Exit Implementation Details	
Document Type	If the user exit can be implemented for a document type, select the appropriate document type, if applicable.
Implement As A Service	If the user exit can be implemented to use a service and you are configuring it as such, choose Implement As A Service.

Table 4–7 User Exit Implementation Details Fields

Field	Description
Implement As A Java Class	If you are configuring the user exit to be implemented as a Java class, choose Implement As A Java Class.
Service Name (if selected Implement as Service)	If you selected Implement As A Service, select the applicable service to configure.
Java Class (if selected Implement as Java Class)	If you selected Implement As A Java Class, enter the Java class as it appears in the User Exit Name field.
Requires Backward Compatibility	Select this field if the user exit requires backward compatibility for another release.
Version	If you selected Requires Backward Compatibility, select the Yantra version number that requires user exit backward compatibility.
Restrict Number Of Calls	If a call made to the external system through the User exit custom code hangs, the API thread, which invoked this user-exit, also hangs. Potentially this could block out all the execute threads in an App server. If checked, you can configure Calls Per JVM, Waiting Calls and Wait time.
Pool Size	Indicates total number of concurrent active calls to User Exit.
Maximum Queue Length	The maximum queue length for the number of user exit calls that will wait to become active if the active count is filled up. If the queue is filled with calls waiting to be active, any new User Exit requests will cause an error.
Wait Time (seconds)	Time for which the user exit call will wait in queue. If the wait time exceeds the configured wait time, an exception will be thrown.
User Exit Implementation Notes	Enter any additional information regarding user exit implementation.

4.2.6.2 Creating an Extended Transaction that is Derived from an Abstract Transaction

You can create new transactions by deriving from existing system transactions in the process type you are working in. These transactions can then be used in pipeline creation and modification.

To create a derived transaction:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Transactions tab.
4. Select the Transactions branch and choose . The Create New Transaction pop-up window is displayed.
5. Select 'Derive from this abstract transaction' and select the applicable transaction to build a derived transaction off of.
6. Choose OK. The Transaction Detail window is displayed in the work area.
7. Enter information in the applicable fields. Refer to [Table 4–3](#) for field value descriptions.
8. Choose .
9. If you modified the Java class, restart the appropriate Agent Servers for the changes to take effect.

4.2.6.2.1 Creating a Status Change Listener Derived Transaction

You can create listener transactions to keep track of the changes in a document when it is in another pipeline. For example, if you are creating an order fulfillment pipeline in which the order document is dropped into an outbound shipment pipeline with its own set of statuses for shipment, you can configure a status change listener transaction in the order fulfillment pipeline to keep track of the statuses the order document goes through in the outbound shipment pipeline.

The following are the abstract status change listener transactions:

- Chained Order Status Listener
- Derived Order Status Listener
- Shipment Status Listener

To configure a status change listener:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Transactions tab.
4. Select the Transactions branch and choose . The Create New Transaction pop-up window is displayed.
5. Select 'Derive from this abstract transaction' and select the applicable listener transaction.
6. Choose OK.
7. In the repository, choose the pipeline tab and then select the pipeline you want to add the status change listener transaction to. Right-click on the pipeline and choose Details. The details of that pipeline appear in the work area.
8. Choose the transactions tab and drag your transaction into the appropriate spot in the work area.
9. Right-click on the status change listener transaction and choose Show Listener Details. The Listener Details pop-up window appears.

Listener Details

Listener Transaction:

Process Type Listening To:

Status Listening To	
Status Name	Status

OK Cancel

10. From Process Type Listening To, select the applicable process type pipeline from which you want the listener to track statuses.
11. From the Status Listening To list, select . A list of statuses that can be tracked is displayed.
12. Select the statuses you want the listener to track and choose .
13. Choose OK.

4.2.6.3 Modifying a Transaction

To modify a transaction:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.

2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Transactions tab.
4. Expand the Transactions branch.
5. Select the applicable transaction and choose . The Transaction Details window is displayed in the work area.
6. Modify information in the applicable fields. Refer to [Table 4–3](#) for field value descriptions.
7. Choose .

4.2.6.4 Deleting a Transaction

To delete a transaction:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Transactions tab.
4. Expand the Transactions branch.
5. Select the applicable transaction and choose .

Note: If a transaction existing in any pipeline is deleted, it appears bright red in the graphical pipeline.

4.2.7 Defining 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.

Yantra 7x provides a default set of statuses. These statuses are used to connect transactions. Your business practices may call for use of one or more extended statuses. These statuses do not stand alone and only follow the status from which they are extended.

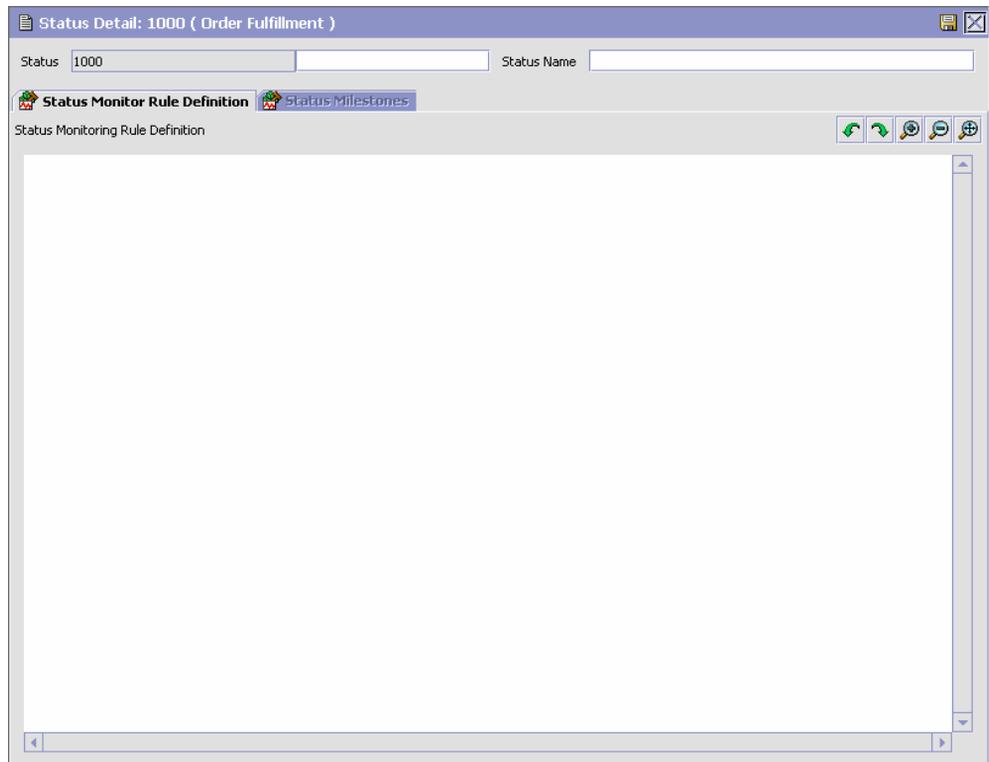
You can use the Statuses tab for:

- [Creating an Extended Status](#)
- [Modifying an Extended Status](#)
- [Deleting an Extended Status](#)
- [Defining Status Monitoring Rule Definitions](#)

4.2.7.1 Creating an Extended Status

To create an extended status:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Statuses Tab.
4. Expand the Statuses branch.
5. Select the applicable status and choose . The Status Detail window is displayed in the work area.



6. In Status, enter the extension number. This number must be sequential with any other existing extended statuses.
7. In Status Name, enter the name of the extended status.
8. Choose .

4.2.7.2 Modifying an Extended Status

To modify an extended status:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.

3. Choose the Statuses Tab.
4. Expand the Statuses branch.
5. Select the applicable extended status and choose . The Status Detail window is displayed in the work area.
6. In Status Name, enter the name of the extended status.
7. Choose .

4.2.7.3 Deleting an Extended Status

To delete an extended status:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Statuses Tab.
4. Expand the Statuses branch.
5. Select the applicable extended status and choose .

4.2.7.4 Defining Status Monitoring Rule Definitions

A status monitoring rule is used to monitor business documents that stay in a particular status for a set amount of time. When the configured time is reached the actions you define in the status monitoring rule definition work area are performed.

To set up status monitoring rule definitions:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Expand the Statuses branch.
4. Choose the Statuses Tab.

5. Double click the applicable Status. The Status Details window is displayed in the work area.
6. Right-click in the work area and select Actions > Add Monitor Node. A monitor node appears in the work area.
7. Drag the applicable actions and conditions into the work area and connect them as per the rules detailed in this section.
8. Connect the status monitor node to the applicable actions. The hours that a document stays in the status before the action is raised is displayed on the connecting line. To change the time, right-click on the time, choose Change, and enter the new time.

Note: Do not set up more than one action for the same monitoring age.

9. Choose .

Note: For the following process types, status monitoring rules cannot be added, and the Status Monitor Rule Definition tab will therefore be disabled:

- Count Execution
 - General
 - Load Execution
 - Manifesting
 - Move Request Execution
 - Outbound Picking
 - Over Pack Build
 - Pack Process
 - Purchase Order Receipt
 - Return Receipt
 - Task Execution
 - Trailer Loading
 - Transfer Order Receipt
 - VAS Process
 - WMS Layout Definition
 - WMS Putaway
 - WMS Inventory
-
-

4.2.8 Defining 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 example, at a certain point in a Sales Order Fulfillment process-type pipeline, you set up a condition to determine if an order contains

hazardous materials. When an order reaches this condition in the pipeline, it cannot move any further until the condition is met with a definitive 'yes' or 'no' value. In this example, if the order contains no hazardous materials, the value is 'no' and the order continues through the regular pipeline. If the order does contain hazardous material, the value is 'yes' and the order is sent down an alternate branch of the order pipeline that has been configured to deal with hazardous material orders.

You can use the Conditions tab for:

- [Creating a Condition](#)
- [Modifying a Condition](#)
- [Deleting a Condition](#)
- [Viewing All Entities Affected by a Condition](#)

4.2.8.1 Creating a Condition

To create a condition:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Conditions Tab.
4. Expand the Conditions branch.
5. Choose . The Condition Details window is displayed in the work area.
6. Enter information in the applicable fields. Refer to [Table 4–8](#) for field level descriptions.
7. Choose .

The screenshot shows a window titled "Condition Detail: New (Order Fulfillment)". It contains the following fields:

- Condition ID: [Text Input]
- Condition Name: [Text Input]
- Condition Group: [Text Input]
- Is Dynamic:
- Condition Value: [Text Input] with a help icon

Table 4–8 Condition Details Window

Field	Description
Condition ID	Enter the condition ID.
Condition Name	Enter the name of the condition.
Condition Group	Enter the name of the condition's group, if applicable. Condition Group allows you to group related conditions within the condition tree.
Is Dynamic	If this is checked you must enter a Java class name that evaluates the condition at runtime.
Class Name (if Is Dynamic is checked)	Enter the class name that implements the following Java interface: <code>com.yantra.ycp.japi.YCPDynamicCondition</code>

Table 4–8 Condition Details Window

Field	Description
Condition Properties (if Is Dynamic is checked)	Specify the custom name or value properties which will be set into the condition evaluating java class file before evaluating the condition. For more information on creating custom attributes see the <i>Yantra 7x Customization Guide</i> .
Condition Value (if Is Dynamic is not checked)	Choose the Condition Builder button to use the condition builder. Here you can use the Condition Builder to set up the condition value. You can set it up in a formulaic readout using the available symbols. You can enter your own attribute or an extended attribute if IsDynamic condition is not checked. For more information on creating these attributes refer to <i>Yantra 7x Customization Guide</i> .

4.2.8.1.1 Using the Condition Builder

You can use the condition builder to create condition values. To use the condition builder you must first select the field(s) to be analyzed when the condition is used and associate the proper value with them.

For example, you want to set up a condition to search for a specific node for order fulfillment, in this example SN1. To set up this condition value, select Ship Node from the list of available order fulfillment fields. From the drop down list select 'Is' and enter SN1 as the value and choose Add. You have now created a condition value that reads "Ship Node is 'SN1'". This indicates that when this condition is used the application checks the document to see if it is associated with SN1, if it is the document moves through the pipeline as per your configuration.

You can also check for conditions to be evaluated if they are greater than, greater than equal to, less than, less than equal to and contains based on the fields you have selected.

You can build more complex strings when creating a condition value using the condition builder. For example, you decide that along with setting up a condition value associated with SN1, you do not want the condition to include any item IDs associated with Item1. To set up this condition value, select Ship Node from the list of available order fulfillment fields. From the drop down list select 'Is' and enter SN1 as the value and choose Add. Then select the statement and choose the open and closed parentheses buttons. After this statement is set up, select

Item ID from the list of available order fulfillment rules. From the drop down list select 'Not Equal To' and enter Item1. Select the statement and choose the & button, then choose the open and closed parentheses buttons. You have now set up a statement to read "(Ship Node Is 'SN1') AND (Item ID Not Equal To 'Item1')". This statement indicates that when this condition is used, the application looks at a given document to see if it is associated with SN1 but not Item1. If this is the case the document goes along the pipeline as per your configuration.

Important: You must use parentheses when using multiple fields in a condition statement.

Important: You can only have two conditions between the bracket symbols.

You can add custom attributes by process types and during condition definition to be evaluated as part of the condition builder functionalities. For more information on implementing custom attributes and incorporating them in the condition builder, see *Yantra 7x Customization Guide*.

4.2.8.2 Modifying a Condition

To modify a condition:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Conditions Tab.
4. Expand the Conditions branch.
5. Expand the applicable condition group branch.

6. Select the applicable condition and choose . The Condition Details window is displayed in the work area.
7. Enter information in the applicable fields. Refer to [Table 4–8](#) the [Transaction Details Window](#) table for field level descriptions. (Refer to [Section 4.2.8.1.1, "Using the Condition Builder"](#) on page 197 for information about modifying a condition that is not dynamic.)
8. Choose .

4.2.8.3 Deleting a Condition

To delete a condition:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Conditions Tab.
4. Expand the Conditions branch.
5. Expand the applicable condition group branch.
6. Select the condition you want to delete and choose .

4.2.8.4 Viewing All Entities Affected by a Condition

You can view all of the events, pipelines, and status rules that are affected by a particular condition. This is useful when you need to modify a condition so that you can see what is impacted by your modification.

To view the entities affected by a condition:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Conditions Tab.
4. Expand the Conditions branch.

5. Expand the applicable condition group branch.
6. Select the applicable condition and choose . The Condition Details window is displayed in the work area.
7. Choose . The Entities Affected by this Condition pop-up window appears. The Pipeline Entities tab provides a list of pipelines affected by the condition, Enterprises affected by determination rules containing the condition, and pipelines affected by monitoring rules containing the condition. The Others tab details all of the events, statuses, and services affected by the condition.

Entities Affected By This Condition

Pipeline Entities Others

Pipeline List

Pipeline ID	Description
SALES-701	Sales Order Fulfillment

Results 1 Of 1

Enterprise List for affected Pipeline Determination Rule

Enterprise

Results 0 Of 0

Pipeline List for affected Monitor Rule

Pipeline ID	Description
-------------	-------------

Results 0 Of 0

4.2.9 Defining Actions

An **action** is a process or program that is triggered by an event. These processes and programs send alert notifications, publish data, or initiate custom services.

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

You can use the Actions tab for:

- [Creating an Action](#)
- [Modifying an Action](#)
- [Deleting an Action](#)

4.2.9.1 Creating an Action

To create an action:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Actions Tab.
4. Expand the Actions branch.
5. Choose . The Action Details window is displayed.
6. Enter information in the applicable fields. Refer to [Table 4–9](#) for field level descriptions.
7. Choose .

Note: It is recommended that all Actions defined by you should contain the prefix "EXTN_" to avoid conflicts between factory-shipped actions and the custom defined actions.

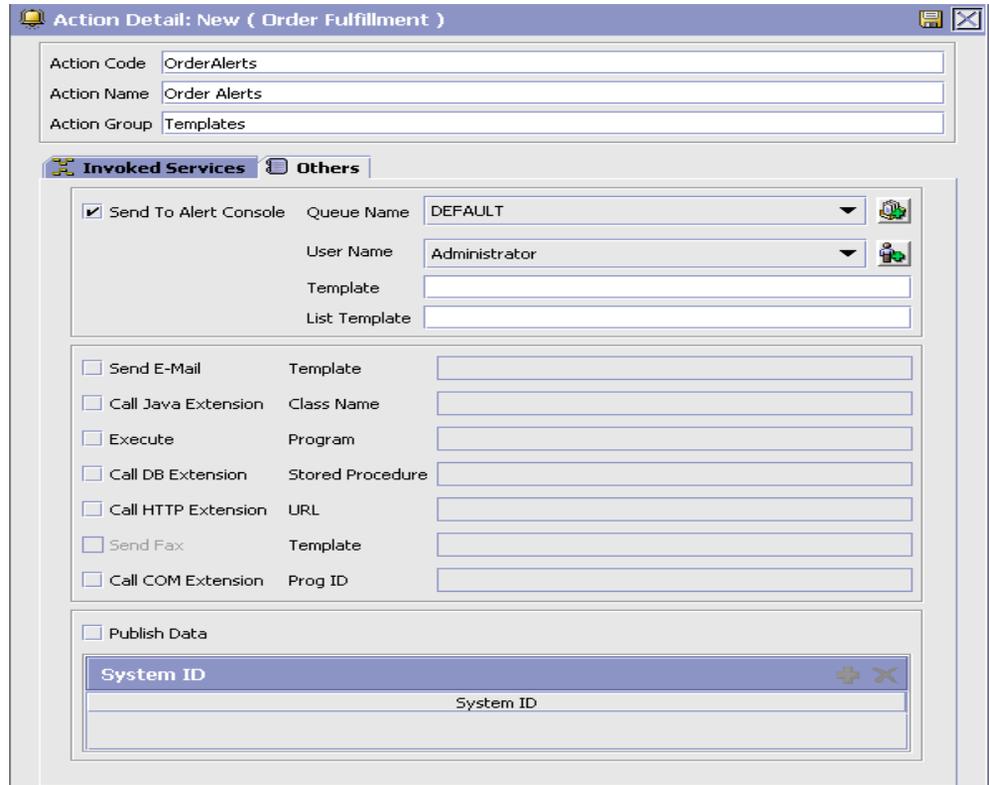


Table 4–9 Action Details Window

Field	Description
Action Code	Enter the action code.
Action Name	Enter the action's name.
Action Group	Enter the name of the action's group, if applicable. Actions belonging to the same group will appear together in the Action tab in the Process Modeling tree.
Invoked Services	

Table 4–9 Action Details Window

Field	Description
Invoke following services as part of this action	<p>Select this if you want this action to invoke a configured service.</p> <p>Note: If you configure actions that invoke a service and that service inserts messages into an MQ Series queue, ensure that your System Administrator includes the following .jar files in the CLASSPATH environment variable for your application server (WebLogic or WebSphere):</p> <pre> \$MQSERIES_HOME/lib/fscontext.jar \$MQSERIES_HOME/lib/providerutil.jar \$MQSERIES_HOME/lib/jndi.jar \$MQSERIES_HOME/lib/com.ibm.mq.jar \$MQSERIES_HOME/lib/com.ibm.mqbind.jar \$MQSERIES_HOME/lib/com.ibm.mqjms.jar \$MQSERIES_HOME/lib/jms.jar </pre>
Invoked Services List	<p>Lists the services that this action invokes. When you check 'Invoke following services as part of this action' you can add additional services by choosing . You can remove services by selecting the applicable service and choosing .</p>
Service Name	The name of the service.
Service Group Name	The name of the service group the service belongs to.
Others	Important: All information in this tab can be configured within the Service Definition Framework. The information in this tab is provided solely for backward compatibility purposes. From Version 5.0 onwards, these should be configured as a service.
Send to Alert Console	Select the check box if you want an alert notice to be sent to a particular user.
Queue Name	Enter the name of the queue the alert should be sent to.
User Name	Select the name of the user who is to receive alert notices.

Table 4–9 Action Details Window

Field	Description
Template	<p>Enter the Alert Console template. It can be any name followed by an ECT or XSL extension.</p> <p>If the template is within the EAR file:</p> <p>The value of the Template field specified in the action should be the same as the path to the template file as built within the EAR. The path should be relative to the root of the EAR.</p> <p>If the template is outside the EAR file:</p> <p>The value of the Template field specified in the action can be the path to the file relative to the path given in the CLASSPATH specified in the application server's start-up script.</p>
Send E-Mail	<p>Select the check box if you want an e-mail message to be sent. Enter a template name in the Template field.</p> <p>Note: You must configure your e-mail server before you can activate this action.</p>
Template	<p>Enter the name of the e-mail template. It can be any name followed by an MLT extension (if data published is a map) or XSL extension (if data published is an XML).</p> <p>If the template is within the EAR file:</p> <p>The value of the Template field specified in the action should be the same as the path to the template file as built within the EAR. The path should be relative to the root of the EAR.</p> <p>If the template is outside the EAR file:</p> <p>The value of the Template field specified in the action can be the path to the file relative to the path given in the CLASSPATH specified in the application server's start-up script.</p>
Call Java Extension	<p>Select this field if you want to call a particular Java component.</p>
Class Name	<p>Enter the Java class name.</p>
Execute	<p>Select this field if you want to call a particular executable.</p>
Program	<p>Enter the Program (executable) name. Make sure the executable exists in the system PATH.</p>

Table 4–9 Action Details Window

Field	Description
Call DB Extension	Select this field if you want to call a particular stored procedure.
Stored Procedure	Enter the Stored Procedure name.
Call HTTP Extension	Select this field if you want to call a particular URL.
URL	Enter the URL to be called for the HTTP Extension.
Send Fax	This field is no longer supported.
Template	This field is no longer supported.
Call COM Extension	This field is no longer supported.
Prog ID	This field is no longer supported.
Publish Data	Select this field if you want to publish data to an external system.
System ID	<p>Choose  and enter the system ID in the System ID pop-up window.</p> <p>If you want to delete an existing system ID, select the system ID you want to delete and choose .</p> <p>Note: System IDs cannot be more than 20 characters.</p>

Note: Only actions linked to the primary enterprise of this node or organization will be available in the dropdown. Any actions created from this screen using the create button will be linked to the primary enterprise of the user's organization and hence may not be available for the node or organization being created.

In a multi-enterprise situation please ensure that actions are created for the appropriate enterprises first (when logged in as that enterprise user).

Subsequently mapping of nodes to actions can be done logged in either as an enterprise user or as a hub user.

4.2.9.2 Modifying an Action

To modify an action:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Actions Tab.
4. Expand the Actions branch.
5. Expand the applicable action group branch.
6. Select the applicable action and choose . The Action Details window is displayed.
7. Enter information in the applicable fields. Refer to [Table 4–9](#) for field value descriptions.
8. Choose .

4.2.9.3 Deleting an Action

To delete an action:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Actions Tab.
4. Expand the Actions branch.
5. Expand the applicable action group branch.
6. Select the applicable action and choose .

4.2.10 Defining Service Definitions

Service definitions are a representation of the logic that regulates document workflow services. The Service Builder is a graphical interface that enables you to create a graphical representation of these *services*, in the following situations:

- Transporting data, typically between Yantra 7x and external applications
- Transforming data from one format to another
- Extending the application logic when events are raised

Services can be accessed using the following mechanisms:

- `executeFlow()` API
- Resource configuration for accessing from the user interface
- Actions can be associated to invoke a service
- User-triggered transactions can be used to invoke a service to raise an alert to inform the applicable users
- Document Routers
- Monitors

Service Nodes

Service nodes contain the logic that you can use to build a service definition.

The following service nodes are available from the Service Palette:

- Transport nodes
- Component nodes
- Adapter nodes
- Connector nodes

Connector nodes are only available from the right-click menu.

Transport Nodes

Transport nodes forward messages, allowing Yantra 7x to communicate with external systems. Transports (and the entire service) can be classified into the following categories:

- Synchronous - immediately forward messages
- Asynchronous - store and forward messages

You may use either type, depending on your needs. The following sections list the types of synchronous and asynchronous transport types.

You can add a transport node by dragging it from the pallet into the work area.

Synchronous services forward messages immediately. Yantra 7x supports the following synchronous transport types:

- COM
- Enterprise Java Bean (EJB)
- Hypertext Transfer Protocol (HTTP)
- Web Services
- Synchronous MQSeries Message Queue
- Synchronous WebLogic Message Queue

Asynchronous services store and forward messages. They queue up messages in a database or a queuing mechanism, which allows you to reprocess exceptions, if any, at a later time. Yantra 7x supports the following asynchronous transport types:

- Asynchronous MQ JMS Queue
- Asynchronous WebLogic JMS Queue
- Database
- File IO
- FTP
- MSMQ

Each transport type has the following sender and receiver aspects:

- receiver - defines how information should be received from the transport node
- sender - defines how information should be sent to the transport

Whether a transport is a sender or receiver depends on how you have connected the flow of logic to be directed.

For a complete list of available transport nodes and details of their parameters, see [Appendix C, "Service Builder Nodes and Parameters"](#).

Component Nodes

Component nodes format or translate data. Yantra 7x supports the following components:

- Alert
- API
- E-Mail
- Composite Service
- Condition
- Nomenclature Runtime
- Router
- Text Translator (For detailed information about text translator file configuration, see the [Appendix D, "Text Translator Reference"](#))
- XSL Translator

You can add a component node by dragging it from the pallet into the work area.

For a complete list of available component nodes and details of their parameters, see [Appendix C, "Service Builder Nodes and Parameters"](#).

Adapter Nodes

Adapter nodes allow you to implement a Yantra 7x Adapter with an external system.

Yantra 7x supports the following adapter:

- [Sterling GIS](#)

For details of this adapter node's parameters, see [Section C.3, "Adapter Nodes"](#).

Connector Nodes

Connector nodes allow you to link nodes together without adding any additional logic. This allows you to complete a service. The types of available connector nodes are as follows:

- Start node - All services are required to begin with a Start node. The Start node defines where to begin running the Yantra 7x Service Definition Framework logic. When you create a new flow, the Start node is already laid out for you.
- End node - All services are required to end with an End node. The End node defines where to end that particular flow of the Yantra 7x Service Definition Framework logic. When you create a new flow, the End node is already laid out for you.
- Pass-through node - The Pass-through node allows you connect synchronous and asynchronous components together.

You can add a connector node by right-clicking in the work area and selecting from the above connector node types.

Criteria of a Complete Service Flow

The following conditions must be met in order to save a service.:

- Start node - Required. One maximum.
- Transport node - Optional. Zero or many.
- Component node - Required. One or many.
- Adapter Node - Optional. Zero or many.
- End Node - Required. One or many.
- All nodes must be connected together.
- All required properties on all nodes and links must have values specified.

You can use the Services tab for:

- [Creating a Service](#)
- [Saving a Service as a Draft](#)
- [Saving a Service as Another Service](#)

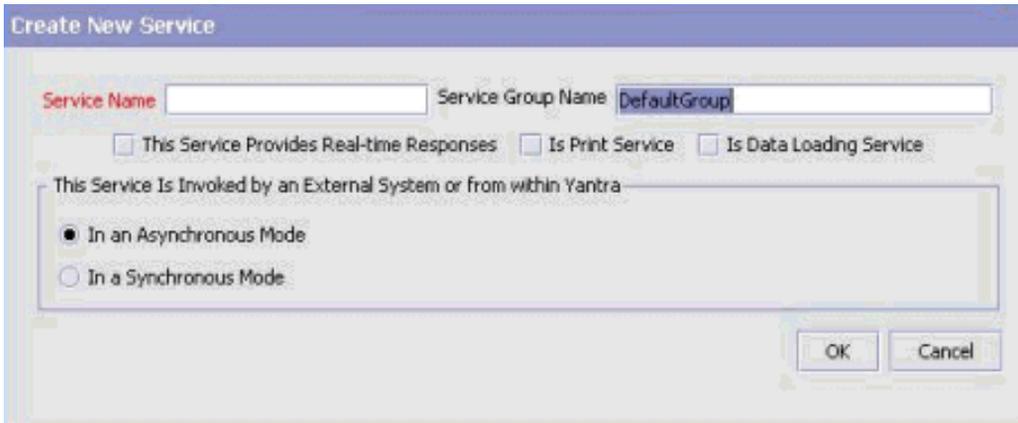
4.2.10.1 Creating a Service

To create a service:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.

2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Service Definitions Tab and select the parent node of the current Service Definition tree.
4. Choose the Process Type Services node and choose . The Create New Service Properties dialog box appears.
5. Enter information in the applicable fields. Refer to [Table 4–10](#) for field value descriptions.
6. Choose OK. The Service Details Window appears.

Note: It is recommended that all Services defined by you should contain the prefix "EXTN_" to avoid conflicts between factory-shipped services and the custom defined services.



Service Name Service Group Name

This Service Provides Real-time Responses Is Print Service Is Data Loading Service

This Service Is Invoked by an External System or from within Yantra

In an Asynchronous Mode
 In a Synchronous Mode

OK Cancel

Table 4–10 Create New Service Window

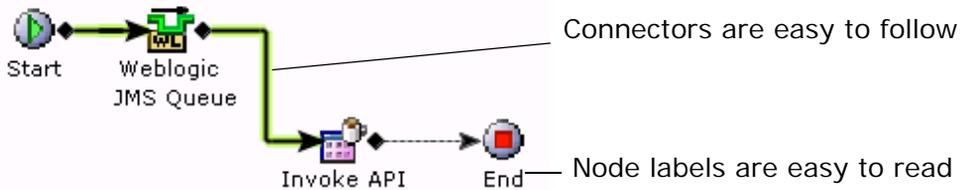
Property	Description
Service Name	Enter the service name.
Service Group Name	Enter the service group the new service should be categorized in. As long as the service group is populated with services, it cannot be deleted. When you delete all services within a group, the group container is automatically deleted.
This Service Provides Real-time Responses	Select this if the service returns a response to the caller when invoked. This option is only available for services that are invoked synchronously, whether from within Yantra 7x or externally. Note: When this property is selected, asynchronous transports cannot be added to the service.
Is Print Service	Select this service to print labels.
Is Data Loading Service	Select this service to load data.
This Service Is Invoked by an External System or from within Yantra 7x	
In an Asynchronous Mode	Select this option when the service must start by retrieving a message from an asynchronous transport source. The service starts from a queue or database. The definition of the service does not need detail how the message arrives at asynchronous source such as queue or database. When this option is selected the first node after the start node must be an asynchronous transport node.
In a Synchronous Mode	Select this option when the service is invoked from Yantra 7x or through an API synchronously.

You can use the Service Work Area for:

- [Linking Service Definition Nodes](#)
- [Defining a Node's Properties](#)

4.2.10.1.1 Linking Service Definition Nodes

Before linking nodes together, lay them out on the work area so that the logic flows from left to right and from top to bottom. Place your nodes so that the link coming from a source node are on the bottom or right side, and comes into the top or left side of the target node. See the following example.



To link nodes together:

1. On the first node, click the small black diamond to its right.
2. Drag a link to the next node.

It doesn't matter what order you use for linking nodes, but it makes more sense to create the links in the order in which the logic flows.

Caution: If you delete a transport link, you also delete any properties you have defined. You can avoid this by linking your nodes together before defining the properties.

If you try to link nodes together that cannot be linked, the status bar informs you that this task cannot be completed.

4.2.10.1.2 Defining a Node's Properties

You can configure the individual node properties. For a complete list of the available service nodes and their properties, see [Appendix C, "Service Builder Nodes and Parameters"](#).

To configure node properties:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.

2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Services Tab and select the parent node of the current Service Definition tree.
4. Choose the Process Type Services node.
5. Locate the applicable service and choose . The Service Details Window appears.
6. In the work area, choose the applicable node, its properties panel appears in the bottom frame.
7. Edit the properties as indicated for the node in [Appendix C, "Service Builder Nodes and Parameters"](#).

4.2.10.2 Saving a Service as a Draft

You can save an incomplete service as a draft. This draft can be retrieved for a final save without any necessary validations.

To save a service as a draft:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Services Tab and select the parent node of the current Service Definition tree.
4. Configure a service as per the rules detailed in this section.
5. Choose . The service is saved as a draft service.
6. When you are ready to save it as a complete and functional service, choose .

Note: When you save a service as a draft, any existing drafts for the service are overwritten. When you save the draft as an actual service, any existing services are overwritten.

4.2.10.3 Saving a Service as Another Service

You can save an existing service as another service.

Note: When you save a service containing a Sub Service Name as another service, that Sub Service Name is copied over with an appended digit to differentiate it from the original Sub Service Name. For example, if you save a service named Service1 containing a Sub Service Name R1 as another service named Service2, the original Sub Service Name is copied over as R1_0.

To save a service as another service:

1. In the Process Modeling window, select the Order, Load, or General tab to view the corresponding process modeling tree for that base document type.
2. In the Process Types swimlane, right-click on the applicable process type and choose Model Process. The Repository Details window and work area are displayed for the corresponding process type.
3. Choose the Services Tab and select the parent node of the current Service Definition tree.
4. Select the existing service you want to save as a new service.
5. Choose . The Save Service As pop-up window is displayed.
6. In Service Name, enter the name of the new service.
7. Choose .

5

Configuring User Security

Security must be set up to allow users access to the actions and views provided by the organization to which they belong. A user is limited to access only those to which they have permission.

Yantra 7x Configurator's **Security Management** is used to create users, user groups, and data security groups. Once these have been created, permissions can be assigned.

You can use the Security branch for:

- [Defining User Groups](#)
- [Defining Users](#)
- [Defining Data Security Groups](#)

5.1 Defining User Groups

User groups are a collection of users who perform a similar task. For example, a group of customer service representatives might be put in a Customer Service Representative user group. Users can belong to multiple user groups to which permissions are assigned. A user who belongs to multiple user groups retains the least restrictive set of permissions defined by the groups they belong to. For example, if a user belongs to a user group that permits them to use the Application Consoles and they also belong to a user group that only permits them to use the Application Consoles and Configurator, the user will have access to both applications.

Each organization has its own user groups. User groups can only contain users for the same organization that the user was created for, except in

the case of a user group created by the Hub organization, which can contain users of any organization.

You can use the Groups branch for:

- [Creating and Modifying a User Group](#)
- [Deleting a User Group](#)

5.1.1 Creating and Modifying a User Group

To create a user group:

1. From the tree in the application rules side panel, choose Security > Groups. The Groups window appears in the work area.
2. Choose . The Group Details window displays.
3. Choose the Primary Info tab.
4. Enter information in the applicable fields. Refer to [Table 5–1](#) for field value descriptions.
5. Choose .

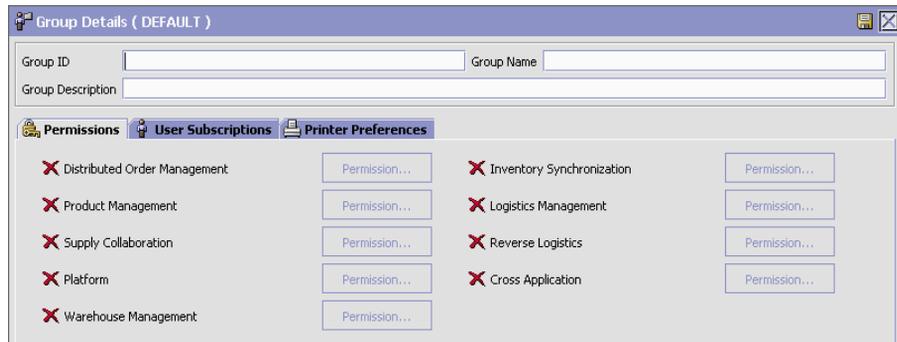


Table 5–1 *Group Details Window*

Field	Description
Group ID	Enter the user group ID.
Group Name	Enter the user group's name.
Group Description	Enter a description about the user group.

To modify a user group:

1. From the tree in the application rules side panel, choose Security > Groups. The Groups window appears in the work area.
2. Select the applicable user group and choose . The Group Details window is displayed.
3. Choose the Primary Info tab.
4. Modify information in the applicable fields. Refer to [Table 5–1](#) for field value descriptions.
5. Choose .

You can also use the Group Details window for:

- [Administering User Group Permissions](#)
- [Viewing the Users Subscribed to a User Group](#)
- [Setting Up Printer Preferences for a User Group](#)

5.1.1.1 Administering User Group Permissions

You can administer the permissions that a user group has throughout the Application Consoles and Configurator applications. You can allow or disallow permissions for an entire module or on a screen-by-screen or function-by-function basis. These permissions apply to all of the users in the user group.

Note: The user administering the permissions is only able to administer permissions for those action and views that he or she has rights to administer. Therefore, it is suggested that each organization have one single user who administers permissions for his or her own organization.

To set up user group permissions:

1. From the Group Details window, choose the Permissions tab.
2. Locate the module that you want to add and/or revoke permissions for and choose the Permission button. The Permissions tree for the corresponding module is displayed.



3. If you want to allow permissions for an entire module, highlight the module you want to allow permissions for and choose the Grant All icon. To disallow permissions for an entire module, highlight the module and choose the Revoke All icon.

You can also view the list of users who have permission to access the entity by performing a right-click and choosing .

Note: If you want to revoke permissions to a particular menu for a given user group, you need to revoke all of the permissions for screens that can be selected under the menu option for which you are revoking permissions. For example, if you uncheck the System Management Console and all of its associated screens and functions, users will not see the System Management Console menu option in the Yantra 7x Application Consoles.

4. If you want to allow permissions on a screen-by-screen or function-by-function basis, expand the application that you want to allow permissions for and highlight the screens that you want to allow and choose the Grant icon. To disallow permissions on a screen-by-screen or function-by-function basis, highlight the screens and choose the Revoke icon.
5. If you are configuring permissions for a group that has access to the Yantra 7x Application Consoles, choose the Cross Application Permission button and expand the Application > Yantra Console > Override branch and enable any of the following permissions as needed:

- The Display Decrypted Primary Payment Attributes permission determines whether sensitive payment information such as credit card number, stored value card number, customer account number or primary payment reference is displayed or masked in the Yantra 7x Application Consoles.

If Yantra 7x is configured to encrypt primary payment attributes and the Display Decrypted Primary Payment Attributes permission is granted, the Yantra 7x Application Consoles will determine whether to call the `getDecryptedString` API to decrypt and display sensitive payment information.

For more information on enabling database encryption for primary payment attributes, refer to [Section 6.3, "Defining Installation Rules"](#) on page 248.

- To grant Yantra 7x Application Consoles users the ability to make modifications on documents that are normally not allowed based on the status modification rules you have configured (reference), grant the Override Modification Rules permission. For example,

you may not allow regular users to modify the instructions of a released sales order. However, specific users should be able to add instructions on exception conditions. When this permission is granted, the user will be able to make the appropriate overriding modifications in the order console.

Note: To indicate that a particular field can be only be modified through this user group permission, the Yantra Console will display this field as editable, with a blue background.

- To grant Yantra 7x Application Consoles the ability to view the stack trace error messages, grant the Display Error Details permission.
6. Choose  after configuring the permissions.

Important: If you are configuring permissions for a group that has access to the Yantra 7x Application Consoles, choose the Cross Application Permission button and expand the Application > Yantra Console > Override branch. Select Display Decrypted Credit Card Number if you want the users in this group to be able to see complete credit card numbers in the Application Consoles. Select Override Modification Rules if you want the permissions that you have configured for this group to override any modification rules that you have configured. Otherwise, leave this box unchecked and the configured modification rules are always applied.

5.1.1.2 Viewing the Users Subscribed to a User Group

You can view all of the users that are subscribed to the user group. The list can be viewed by choosing the User Subscriptions tab from the Group Details window. You can also add or delete a user from the group subscription.



User ID	User Name	Organization
jdoe	John Doe	DEFAULT

5.1.1.3 Setting Up Printer Preferences for a User Group

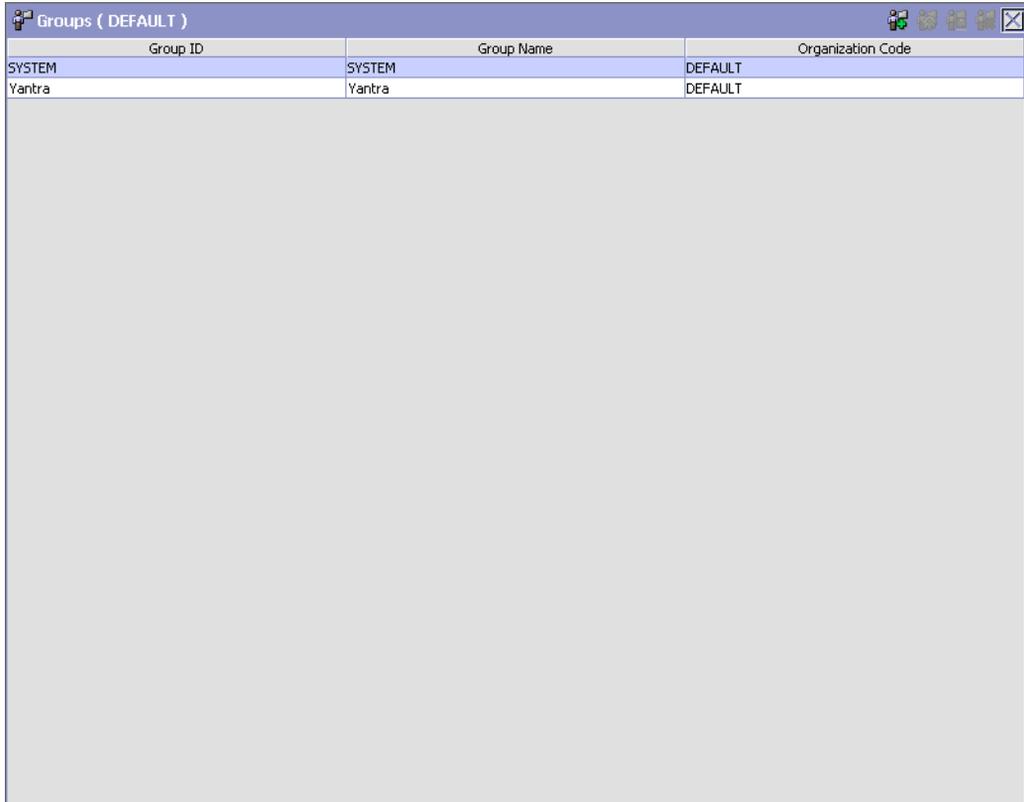
User Printer Preferences configures printers that are associated with a group of users. This preference is used to determine the printer to use when a user prints a document.

For example, receiving office associates all its users to the HP LaserJet 5P located in the office.

The association of a printer to a station overrides the group preference of the specified user. The station is a static location where devices may be directly attached to a station.

It is recommended that User Printer Preferences be configured at the group level for easier administration.

1. From the menu bar, choose Applications > Platform. The Platform tree is displayed in the side panel.
2. From the tree in the application rules side panel, choose Security > Groups. The Groups window is displayed with a list of groups.



Group ID	Group Name	Organization Code
SYSTEM	SYSTEM	DEFAULT
Yantra	Yantra	DEFAULT

3. In the Groups window, choose the Group whose Printer Preferences are to be set up.
4. Choose . The Group Details window is displayed.
5. In the Group Details window, choose the Printer Preferences tab. The Printer Preferences tab window is displayed.
6. Enter the information in the applicable fields. Refer [Table 5–2](#) for field value descriptions.
7. Choose .

For more information about Setting Up a Group (Creating, Modifying, or Deleting a Group), [Section 5.1, "Defining User Groups"](#) on page 221.

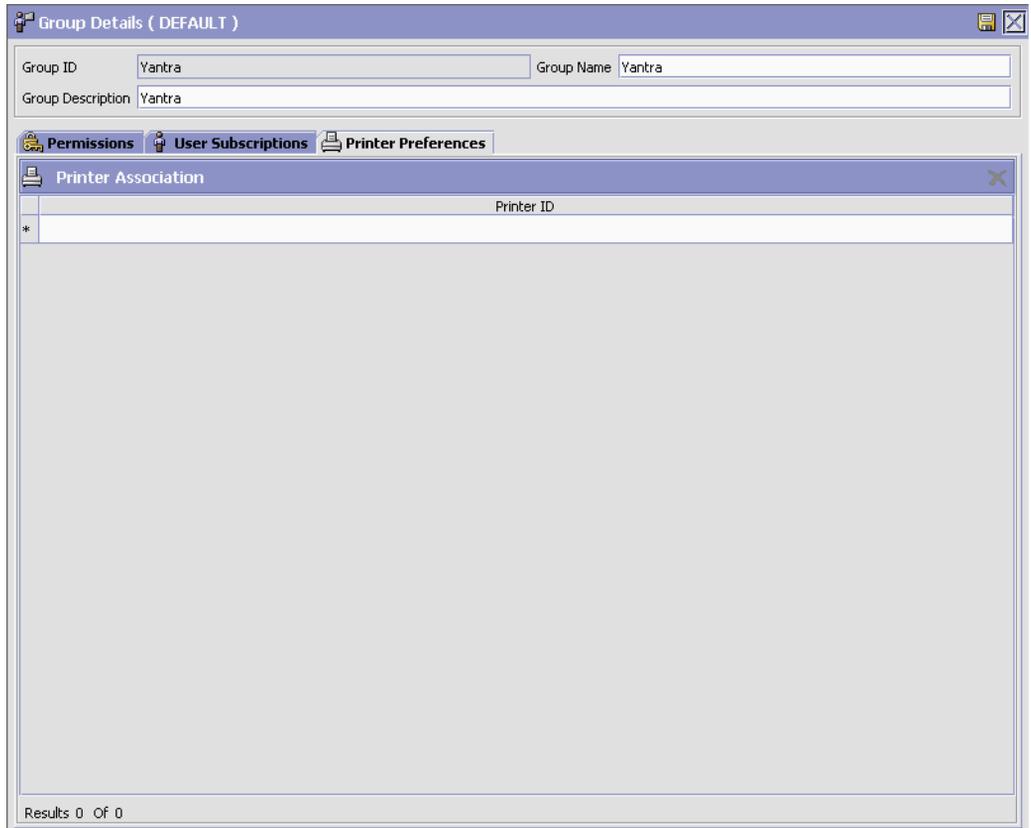


Table 5–2 Printer Preferences Tab Window

Field	Description
Printer Association	
Printer ID	From the drop down, select the printer ID to be associated with the group.

The printer at the packing station is associated to the station and not to the packing group or the individual packer. This is also recommended for warehouses that have only a single pack station.

5.1.2 Deleting a User Group

To delete a user group:

1. From the tree in the application rules side panel, choose Security > Groups. The Groups window appears in the work area.
2. Select the applicable user group and choose .

5.2 Defining Users

A **user** is a single person assigned with a certain task, such as, Hub Administrator or Customer Service Representative, depending on what role they play in the organization. Each user is associated with one organization.

You can use the Users branch for:

- [Creating and Modifying a User](#)
- [Setting Up Printer Preferences for a User](#)
- [Deleting a User](#)

5.2.1 Creating and Modifying a User

To create a user:

1. From the tree in the application rules side panel, choose Security > Users. The User Search window appears in the work area.
2. Choose . The User Details window is displayed.
3. Choose the Primary Info tab.
4. Enter information in the applicable fields. Refer to [Table 5–3](#) for field value descriptions.

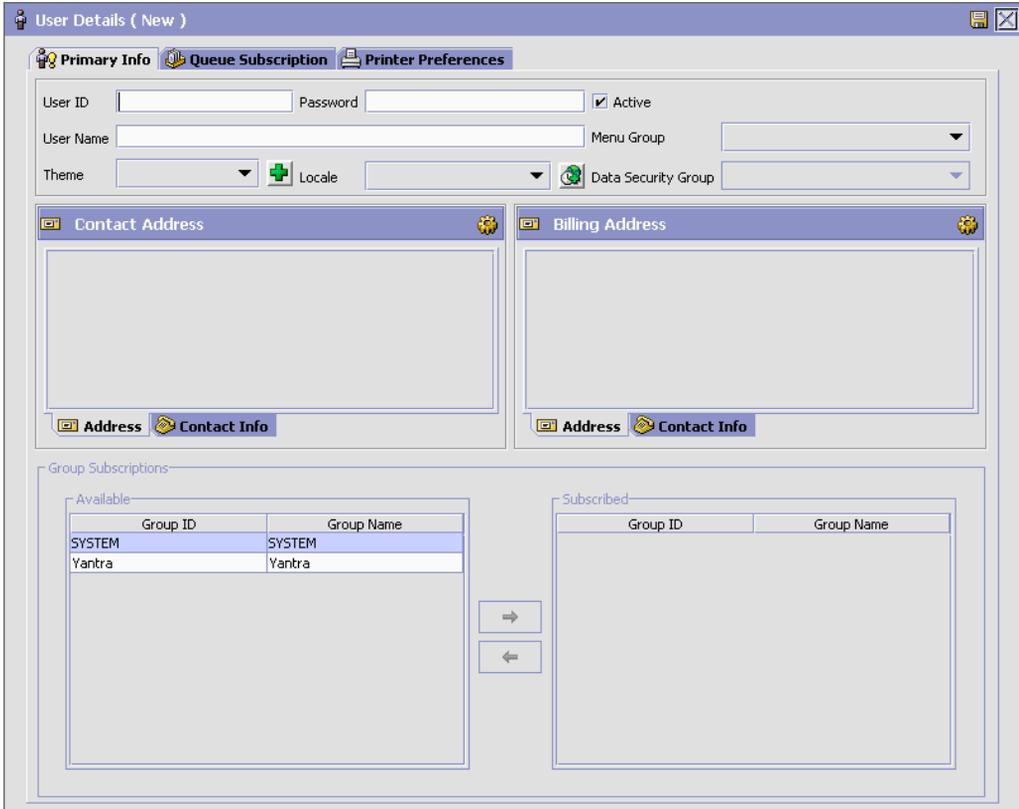


Table 5–3 *User Details Window*

Field	Description
User ID	Enter the user ID that the user uses to access the system.
Password	Enter the password the user uses to access the system.
Active	Check this if the user is currently active in the organization. Inactive users cannot log in.
User Name	Enter the user’s name.
Menu Group	Select the menu group representing the menu options you want this user to see when they log into the Yantra 7x Application Consoles.

Table 5–3 User Details Window

Field	Description
Theme	<p>Enter the available theme as you want it to appear for the user. The theme determines how the color scheme of the Console and Configurator appears to the user.</p> <p>The available themes are:</p> <ul style="list-style-type: none"> • Earth • Jade • Sapphire <p>Note: You can extend the system to include as many themes as you want.</p>
Locale Code	<p>Select the locale the user is located in.</p> <p>Note: A user who is configured for the Eastern Time Zone but logs in while physically in the Pacific Time Zone, sees locale specific information as if he or she was in the Eastern Time Zone.</p>
Data Security Group	<p>Select the data security group you want to associate with the user. For more information about data security groups, see Section 5.3, "Defining Data Security Groups" on page 237</p>
Contact Address	<p>The user's contact address.</p> <p>Choose  to enter an address.</p> <p>Choose the Contact tab to view additional contact information.</p>
Billing Address	<p>The user's billing address.</p> <p>Choose  to enter an address.</p> <p>Choose the Contact tab to view additional contact information.</p>
Group Subscriptions	

Table 5–3 User Details Window

Field	Description
Available	A list of the available user groups. To subscribe a user to a user group, select the applicable user group and choose  .
Subscribed	A list of the user groups to which the user is subscribed. To remove a user from a user group, select the applicable user group and choose  .

5. If you want to subscribe the user to an alert queue or remove the user from an alert queue, choose the Queue Subscription tab.
6. To subscribe a user to an alert queue, select the applicable queue from Available and choose . To remove a user from a queue select the applicable queue from Subscribed and choose .
7. Choose .

To modify a user:

1. From the menu bar, choose Applications > Platform. The Platform tree is displayed in the side panel.
2. From the Platform tree, choose Security > Users. The User Search window appears in the work area.
3. Enter applicable search criteria and choose . A list of users displays.
4. Select the applicable user and choose .
5. Modify information in the applicable fields. Refer to [Table 5–3](#) for field value descriptions.
6. Choose .

5.2.2 Setting Up Printer Preferences for a User

User Printer Preferences configures printers that are associated with a specific user. This preference is used to determine the printer to use when a user prints a document.

For example, receiving office associates all its users to the HP LaserJet 5P located in the office.

The association of a printer to a station overrides the group preference of the specified user. The station is a static location where devices may be directly attached to a station.

It is recommended that User Printer Preferences be configured at the group level for easier administration.

To set up printer preferences for a user:

1. From the menu bar, choose Applications > Platform. The Platform tree is displayed in the side panel.
2. From the tree in the application rules side panel, choose Security > Users. The User Search window is displayed.
3. In the User Search window, enter applicable search criteria.
4. Choose . The list of users is displayed in the Search Results panel of the User Search window.

User Search (DEFAULT)

User ID

User Name

User Group Queue

Status Active Inactive Either Max Records

Search Results

User ID	User Name
doc	doc
doc1	doc1
jdoe	John Doe
roon	roon
yantra	Administrator

Results 5 Of 5

5. In the Search Results panel of the User Search window, choose the User whose Printer Preferences are to be set up.
6. Choose . The User Details window is displayed.
7. In the User Details window, choose the Printer Preferences tab. The Printer Preferences tab window is displayed.
8. Enter the information in the applicable fields. Refer [Table 5–4](#) for field value descriptions.
9. Choose .

For more information about Setting Up a User (Creating, Modifying, or Deleting a User), see [Section 5.2, "Defining Users"](#) on page 230.

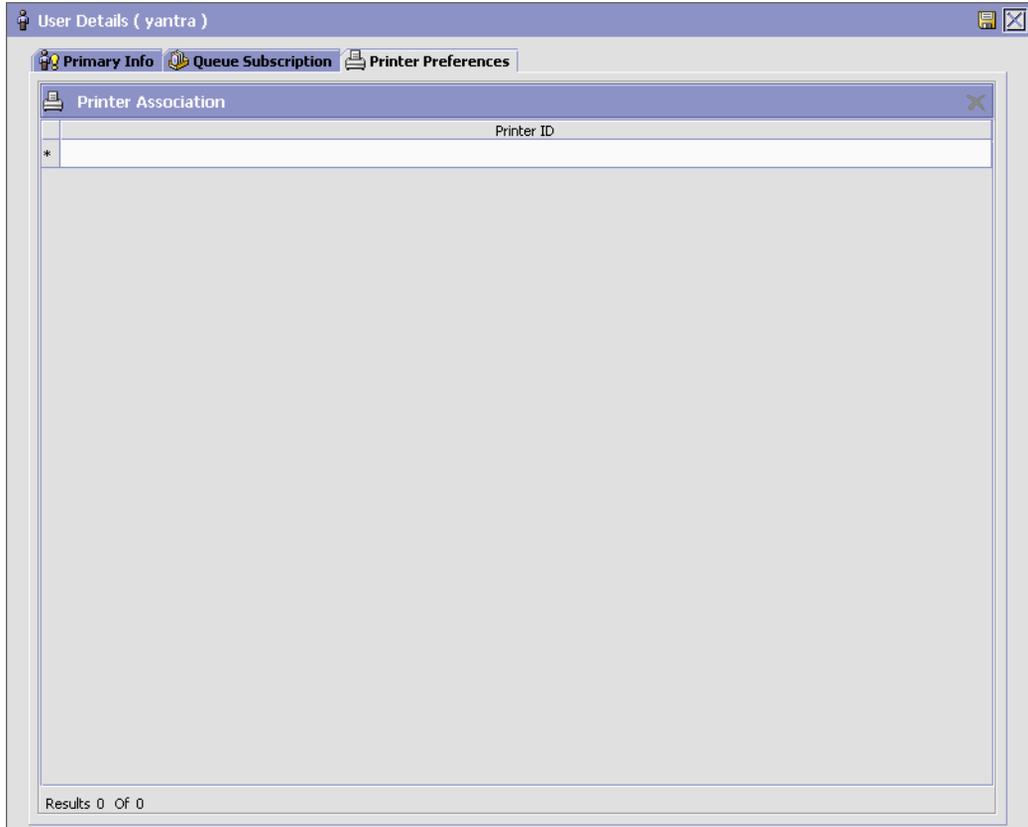


Table 5–4 Printer Preferences Tab Window

Field	Description
Printer Association	
Printer ID	From the drop down, select the printer ID to be associated with the user.

5.2.3 Deleting a User

To delete a user:

1. From the tree in the application rules side panel, choose Security > Users. The User Search window appears in the work area.
2. Enter applicable search criteria and choose . A list of users is displayed.
3. Select the applicable user and choose .

5.3 Defining Data Security Groups

Data security groups are used to control access to the data contained in specific document types and Enterprises within the Yantra 7x Console. Creating a data security group is an optional process. If a user is not associated with a data security group, that user is considered to have the least restrictive access, or default access. By defining a data security group, you can further restrict the access to any Enterprises or document types that are a sub-set of the default access list.

The default access list for document types is based on the document types associated with a particular business module.

The default access list for Enterprise access is based on the type of user for which the data security group is defined. Only Hub, Enterprise, and Enterprise/Hub Node users can create data security groups, as described in the following table:

Table 5–5 Default Access List

User	Default Enterprise Access List	Can be Further Filtered by Data Access Group?
Hub	All Enterprises in the system.	Yes
Enterprise	All Enterprises that are children Enterprises of the Enterprise you are configuring.	Yes
Enterprise/Hub Node Users	All Enterprises with which the node organization participates.	Yes
All Other Users	All Enterprises with which the organization participates.	No

You can define a data security component for data security groups in the service definition framework to enable the groups to secure data based on your data security policies. For more information on data security components, see [Section C.2.11, "Data Security"](#).

You can use the Data Security Groups branch for:

- [Creating a Data Security Group](#)
- [Modifying a Data Security Group](#)
- [Deleting a Data Security Group](#)

5.3.1 Creating a Data Security Group

To create a data security group:

1. From the tree in the application rules side panel, choose Security > Data Security Groups. The Data Security Groups window appears in the work area.
2. Choose . The Data Security Details window is displayed.
3. Enter information in the applicable fields. Refer to [Table 5–6](#) for field value descriptions.
4. Choose .

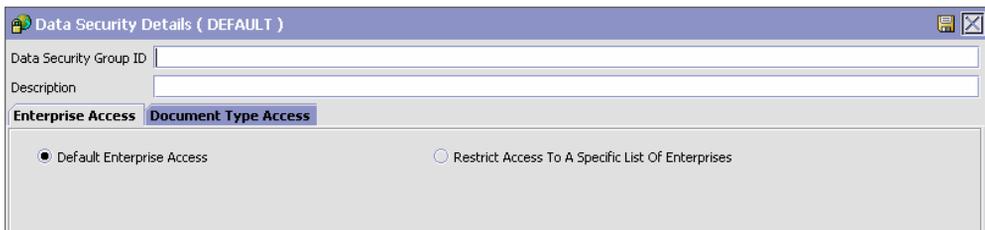


Table 5–6 *Data Security Details Window*

Field	Description
Group ID	Enter the data security group name.
Description	Enter a brief description of the data security group.
Enterprises	

Table 5–6 Data Security Details Window

Field	Description
Default Enterprise Access	Select Default Enterprise Access to restrict users who belong to this data security group to be able to only view the Enterprises belonging to the default Enterprise access list.
Restrict Access To A Specific List Of Enterprises	<p>Select Restrict Access To A Specific List Of Enterprises if you want to create a list of Enterprises that users associated to the data security group can view.</p> <p>Choose  from the Accessible Enterprises table and search for the applicable Enterprises in the Organization Search pop-up window. Choose  to add an Enterprise.</p> <p>Choose  to remove an Enterprise from the Accessible Enterprises list.</p> <p>Note: The restricted list will always include the primary Enterprise of the organization you are configuring the data security group for.</p>
Document Types	
Default Document Type Access	Select Default Document Type Access to allow this data security group to only be able to view the document types belonging to the default access list. The default access list is based on the document types associated with a particular business application.
Restrict List To A Specific List Of Document Types	<p>Select Restrict List To A Specific List Of Document Types if you want to specifically determine the document types the data security group has access to.</p> <p>When you select this option select the business application whose associated document types you want to set access rights to from Applications.</p> <p>Then select if you want to allow Access To All Document Types For This Application or Restrict Access To A Specific List Of Document Types. If you select the latter, the Accessible Document Types table appears displaying all of the accessible document types available for the business application. Select the Accessible check boxes for the applicable document types.</p>

5.3.2 Modifying a Data Security Group

To modify a data security group:

1. From the tree in the application rules side panel, choose Security > Data Security Groups. The Data Security Groups window appears in the work area.
2. Select the applicable data security group and choose . The Data Security Details window is displayed.
3. Enter information in the applicable fields. Refer to [Table 5–6](#) for field value descriptions.
4. Choose .

5.3.3 Deleting a Data Security Group

To delete a data security group:

1. From the tree in the application rules side panel, choose Security > Data Security Groups. The Data Security Groups window appears in the work area.
2. Select the applicable data security group and choose .

6

Configuring System Administration Components

You can configure system level information including system level purge criteria, user exit implementations, and installation rules.

You can use the System Administration branch for:

- [Defining Purge Criteria](#)
- [Defining User Exit Implementations](#)
- [Defining Installation Rules](#)
- [Defining Agent Criteria Groups](#)
- [Defining Initial Context Factory Codes](#)
- [Defining Health Monitor Rules](#)
- [Viewing the List of Configured Servers](#)

6.1 Defining Purge Criteria

You can define purge criteria rules for data purges not related to specific document types. Purges are 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.

[Table 6–1](#) lists the system purge rules provided with Yantra 7x.

Table 6–1 System-Defined Purge Rules

Rule	Description	Retention Days
EXPORTTBLPRG	Purges data from export tables that are used for publishing data to external systems. This is a Hub level purge.	30
REPROCESSPRG	Purges any reprocessed information. This is a Hub level purge.	30
MANIFESTPRG	Purges manifest information.	30
IMPORTTBLPRG	Purges data from the import tables. This is a Hub level purge.	30
INBOXPRG	Purges the Alert Console messages from the user’s inbox. This is a Hub level purge.	30
USERACTIVITYPRG	Purges all user activity data.	30
INVENTORYPRG	Purges inventory information. The inventory purge does not take retention days into account when purging. All records with relevant tables with a quantity of 0 are purged.	30
CAPACITYPRG	Purges capacity data.	30
PRICELISTPRG	Purges price lists.	30
STATBLPRG	Purges statistical information. This is a Hub level purge.	30

6.1.1 Modifying a System Purge Criteria Rule

To modify a purge criteria rule:

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

3. Enter information in the applicable fields. Refer to [Table 6–2](#) for field value descriptions.
4. Choose .

Table 6–2 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 6–2 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	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 Customization Guide</i> . For information about filename limitations related to internationalization, see the <i>Yantra 7x Localization Guide</i> .

6.2 Defining User Exit Implementations

User exits are created to enable business logic extensions to Yantra 7x transactions. Within Yantra 7x transactions, code exists to invoke user exits so that you may plug-in custom logic. Since these are pre-defined by Yantra, you cannot add or delete user exits. However, you can configure appropriate implementations for a user exit.

User exits are Java interfaces which can be implemented for creating custom logic components. Once implemented, they must be configured so that Yantra 7x transactions can invoke them to perform necessary logic at runtime. This chapter explains how to configure user exit implementations within Yantra 7x.

Note: If you do not require Yantra 7x transaction extension, you do not need to implement user exits. If a user exit is not configured, Yantra 7x executes its default business logic. User exits are not relevant when writing custom transactions.

User Exits and Document Types

Document types are a mechanism through which you can manage various business documents and their life cycle. For more information, see [Section 4.1, "Document Type Configuration"](#) on page 131.

Sometimes you need different implementations for a user exit depending

on the document type. For example, the `YFSRecalculateHeaderTaxUE` user exit allows you to compute order header taxes using custom logic. If you want your tax computation logic to differ for Sales Order, Purchase Order, Return, and so on, you can provide different implementations for the same user exit at the document type level. Notice that not all user exits are document type dependent.

User Exits and Services

User exits that take XML input and return XML output are service enabled. This means that for these user exits, instead of writing Java implementations, you can simply attach a service built through the service builder. At runtime, instead of invoking the Java class, Yantra 7x transactions invoke the configured service. This allows a mechanism to build user exit logic in a more declarative fashion than programmatic.

Guidelines for Usage of User Exits

The following guidelines has to be kept in mind when you are using User Exits within Yantra 7x API:

- User Exits are structured to return specific information and hence their usage must be restricted for such purpose alone.
- You cannot call Yantra 7x APIs from an User Exit. For example, you cannot call `changeOrder()` API from a user exit.

6.2.1 Defining a User Exit

To configure user exit implementations:

1. From the tree in the application rules side panel, choose System Administration > User Exit Management. The User Exit List window appears in the work area.

User Exit Name	Can Override For Document Type	Can Attach Service	User Exit Implemented
com.yantra.ycs.japi.ue.YCSOpenMani...	N	Y	N
com.yantra.ycs.japi.ue.YCSCarrierAp...	N	Y	N
com.yantra.ycs.japi.ue.YCSCarrierAp...	N	Y	N
com.yantra.ycs.japi.ue.YCSCloseMani...	N	Y	N
com.yantra.ycs.japi.ue.YCSCloseMani...	N	Y	N
com.yantra.ycs.japi.ue.YCSDeleteCar...	N	Y	N
com.yantra.ycs.japi.ue.YCSDeleteCar...	N	Y	N
com.yantra.yfs.japi.ue.YFSDetermine...	Y	N	N
com.yantra.ycs.japi.ue.YCSGetCode...	N	Y	N
com.yantra.ycs.japi.ue.YCSGetFreigh...	N	Y	N
com.yantra.ycs.japi.ue.YCSGetServic...	N	Y	N
com.yantra.ycs.japi.ue.YCSGetSpecia...	N	Y	N
com.yantra.ycs.japi.ue.YCSGetSpecia...	N	Y	N
com.yantra.ycs.japi.ue.YCSGetTimein...	N	Y	N
com.yantra.ycs.japi.ue.YCSGetTracki...	N	Y	N
com.yantra.ycs.japi.ue.YCSPLDUploa...	N	Y	N
com.yantra.ycs.japi.ue.YCSPLDUploa...	N	Y	N
com.yantra.ycs.japi.ue.YCSShipCarto...	N	Y	N
com.yantra.ycs.japi.ue.YCSShipCarto...	N	Y	N
com.yantra.ycs.japi.ue.YCSShipCarto...	N	Y	N
com.yantra.ycs.japi.ue.YCSTraceShip...	N	Y	N
com.yantra.ycs.japi.ue.YCSTraceShip...	N	Y	N
com.yantra.ydm.japi.ue.YDMBeforeC...	N	Y	N
com.yantra.ydm.japi.ue.YDMBeforeC...	N	Y	N
com.yantra.ydm.japi.ue.YDMBeforeP...	N	Y	N
com.yantra.ydm.japi.ue.YDMBeforeP...	N	Y	N
com.yantra.ydm.japi.ue.YDMDetermin...	N	Y	N
com.yantra.ydm.japi.ue.YDMShouldS...	N	Y	N
com.yantra.ydm.integration.ycs.japi...	N	Y	N
com.yantra.ydm.integration.ycs.japi...	N	Y	N
com.yantra.ydm.integration.ycs.japi...	N	Y	N
com.yantra.ydm.japi.ue.YDMOverrid...	N	Y	N
com.yantra.ydm.japi.ue.YDMBeforeD...	N	Y	N
com.yantra.ydm.japi.ue.YDMBeforeD...	N	Y	N
com.yantra.ydm.japi.ue.YDMCanAdd...	N	Y	N
com.yantra.ydm.japi.ue.YDMGetMoni...	N	Y	N

Results 140 Of 140

If the user exit can be implemented for a document type, the 'Can Override For Document Types' column displays 'Y'. If the user exit can be implemented for services, the 'Can Attach Service' column displays 'Y'. If the user exit is implemented, the 'User Exit Implemented' column displays 'Y'.

2. Locate the applicable user exit and choose . The User Exit Details window appears.
3. From the User Exit Implementation List table, choose . The User Exit Implementation Details appears. Enter information into the applicable fields. Refer to [Table 6–3](#) for field value descriptions.

The screenshot shows a dialog box titled "User Exit Implementation Details". It has a blue header bar with a save icon. The main area is light gray and contains several controls:

- Document Type:** A dropdown menu.
- Implement As a Service:** A radio button.
- Implement As a Java Class:** A radio button, which is selected.
- Requires Backward Compatibility:** A checkbox.
- Restrict Number Of Calls:** A checkbox.
- Pool Size:** A text input field.
- Maximum Queue Length:** A text input field.
- Wait Time (seconds):** A text input field.
- User Exit Implementation Notes:** A large text area at the bottom.

Table 6–3 User Exit Implementation Details Fields

Field	Description
User Exit Implementation Details	
Document Type	If the user exit can be implemented for a document type, select the appropriate document type, if applicable.
Implement As A Service	If the user exit can be implemented to use a service and you are configuring it as such, choose Implement As A Service.
Implement As A Java Class	If you are configuring the user exit to be implemented as a Java class, choose Implement As A Java Class.
Service Name (if selected Implement as Service)	If you selected Implement As A Service, select the applicable service to configure. Important: Only services defined to return a real-time response can be selected. For more information about services, see Section 4.2.10, "Defining Service Definitions" on page 208.
Java Class (if selected Implement as Java Class)	If you selected Implement As A Java Class, enter the Java class as it appears in the User Exit Name field.
Requires Backward Compatibility	Select this field if the user exit requires backward compatibility for another release.

Table 6–3 User Exit Implementation Details Fields

Field	Description
Version	If you selected Requires Backward Compatibility, select the Yantra version number that requires user exit backward compatibility.
Pool Size	Indicates total number of concurrent active calls to User Exit.
Maximum Queue Length	The maximum queue length for the number of user exit calls that will wait to become active if the active count is filled up. If the queue is filled with calls waiting to be active, any new User Exit requests will cause an error.
Wait Time (seconds)	Time for which the user exit call will wait in queue. If the wait time exceeds the configured wait time, an exception will be thrown.
User Exit Implementation Notes	Enter any additional information regarding user exit implementation.

6.3 Defining Installation Rules

You can set up rules that need to be defined when the Hub installs the application.

To set up installation rules:

1. From the tree in the application rules side panel, choose System Administration > Installation Rules. The Installation Rules window appears in the work area.
2. Enter information in the applicable fields. Refer to [Table 7–16](#) for field value descriptions.
3. Choose .

Table 6–4 *Installation Rules Window*

Field	Description
Installation	
Bar Codes Generated on Server	If checked, the system automatically generates bar codes for all items in the Hub environment.
Encrypt Primary Payment Attributes for the following Payment Type Groups	

Table 6–4 Installation Rules Window

Field	Description
Credit Card	If checked, Yantra 7x automatically encrypts the credit card number in the database. Only users with the appropriate permissions can view decrypted credit card numbers. For more information on configuring permissions, refer to Section 5.1.1.1, "Administering User Group Permissions" on page 223.
Stored Value Card	If checked, Yantra 7x automatically encrypts the SVC number, and payment reference 1 in the database. Only users with the appropriate permissions can view the decrypted SVC number and payment reference 1. For more information on configuring permissions, refer to Section 5.1.1.1, "Administering User Group Permissions" on page 223.
Customer Account	If checked, Yantra 7x automatically encrypts the customer account number and payment reference 1 in the database. Only users with the appropriate permissions can view the decrypted customer account number and payment reference 1. For more information on configuring permissions, refer to Section 5.1.1.1, "Administering User Group Permissions" on page 223.
Other	If checked, Yantra 7x automatically encrypts the payment reference 1 in the database. Only users with the appropriate permissions can view the decrypted payment reference 1. For more information on configuring permissions, refer to Section 5.1.1.1, "Administering User Group Permissions" on page 223.
Inventory Consolidation Level	
Hub	Select the Hub option if you want product item IDs to remain unique across all organizations. Important: When an organization is created and this option is selected, the inventory organization will be set to the default Hub organization. This can be changed in the Organization Details screen for the newly created organization. For more information about creating a new organization, see Section 3.1, "Creating and Modifying an Organization" on page 31.

Table 6–4 Installation Rules Window

Field	Description
Enterprise	<p>Select the Enterprise option to allow product item IDs to repeat across enterprises but still be distinguished within the inventory model.</p> <p>Important: When creating an Enterprise organization an this option is selected, the inventory organization will be defaulted to the Enterprise organization you are creating. This can be changed in the Organization Details screen for the newly created organization. For more information about creating a new organization, see Section 3.1, "Creating and Modifying an Organization" on page 31.</p> <p>Note: If you choose Enterprise, an organization or Enterprise code must be specified when performing an inventory adjustment in the Consoles.</p>
Capacity Consolidation Level	
Hub	<p>Select the Hub option if you want service item IDs to remain unique across all organizations.</p> <p>Important: When an organization is created and this option is selected, the inventory organization will be set to the default Hub organization. This can be changed in the Organization Details screen for the newly created organization. For more information about creating a new organization, see Section 3.1, "Creating and Modifying an Organization" on page 31.</p>
Enterprise	<p>Select the Enterprise option to allow service item IDs to repeat across enterprises but still be distinguished within the inventory model.</p> <p>Important: When creating an Enterprise organization an this option is selected, the inventory organization will be defaulted to the Enterprise organization you are creating. This can be changed in the Organization Details screen for the newly created organization. For more information about creating a new organization, see Section 3.1, "Creating and Modifying an Organization" on page 31.</p>
Catalog Model	

Table 6–4 Installation Rules Window

Field	Description
Maintained by Hub	<p>Select the Maintained by Hub option if you want the Hub to set up the a catalog for all of the participants.</p> <p>Important: When an organization is created and this option is selected, the catalog organization will be set to the default Hub organization. This can be changed in the Organization Details screen for the newly created organization. For more information about creating a new organization, see Section 3.1, "Creating and Modifying an Organization" on page 31.</p>
Maintained by Enterprise	<p>Select the Maintained by Enterprise option if you want the individual Enterprises to set up the catalog for all of the participants involved with them.</p> <p>Important: When an organization is created and this option is selected, the catalog organization will be defaulted to the organization's primary Enterprise. This can be changed in the Organization Details screen for the newly created organization. For more information about creating a new organization, see Section 3.1, "Creating and Modifying an Organization" on page 31.</p>
Maintained by Participant	<p>Select the Maintained by Participant option if you want each individual participant to maintain their own catalog.</p> <p>Important: When an organization is created and this option is selected, the catalog organization will be defaulted to the organization itself. This can be changed in the Organization Details screen for the newly created organization. For more information about creating a new organization, see Section 3.1, "Creating and Modifying an Organization" on page 31.</p>
Communication	
E-mail Server Name	Enter the name of the Hub's e-mail server.
E-mail Protocol	Enter the Hub's e-mail protocol.
E-mail Server IP Address	Enter the Hub's e-mail server IP address.
E-mail Server Listener Port	Enter the Hub's e-mail server listener port.
Backward Compatibility	

Table 6–4 Installation Rules Window

Field	Description
<p>Use Deprecated Order Hold Functionality</p>	<p>Check this if you want to use the deprecated order hold functionality instead of the enhanced order hold functionality.</p> <p>For more information on the enhanced order hold functionality, refer to the <i>Yantra 7x Distributed Order Management Configuration Guide</i>.</p> <p>In the deprecated mode, the order hold functionality works as follows.</p> <p>Putting an order on hold freezes the order at its current status in the sales order fulfillment pipeline. You can place an order on hold for any reason. For example, you may want to perform a security check on a particular Buyer, therefore you place the order on hold until you clear the necessary information.</p> <p>The following transactions are not processed when an order is put on hold:</p> <ul style="list-style-type: none"> • Plan Order Complete • Allocate & Release • Chained Order Create • Derived Order Create • Schedule Order • Release Order • Send Release <p>Note: Held orders are not picked up for scheduling.</p> <p>Note: If a shipment is created from an order and the order is then put on hold, the order is still confirmed during shipment confirmation.</p>
Ship Advice Number Generation	
<p>Use Numeric Ship Advice Number</p>	<p>Select this option to use generated numeric values as ship advice numbers.</p>

Table 6–4 *Installation Rules Window*

Field	Description
Maximum Length Of Ship Advice Number	<p>Indicate the maximum length of generated numeric ship advice numbers. Upon reaching the maximum value, the generation sequence is reset. The default value is 9, the maximum length allowable.</p> <p>Note: Due to this constraint, ship advice numbers are not guaranteed to be unique.</p>
Inventory Tag	
Summarize and Maintain Total Supply and Demand Values For Tag Controlled Item	<p>Check this checkbox to summarize and maintain total supply and demand values for tag controlled items.</p> <p>This will update tag descriptors and total supply and demand fields of inventory tags for tag-controlled items during inventory updates.</p>

6.4 Defining Agent Criteria Groups

You can use the Agent Criteria Groups branch for:

- [Creating an Agent Criteria Group](#)
- [Modifying an Agent Criteria Group](#)
- [Deleting an Agent Criteria Group](#)

6.4.1 Creating an Agent Criteria Group

To create an agent criteria group:

1. From the menu bar, choose Applications > Platform. The Platform tree is displayed in the side panel.
2. From the Platform tree, choose System Administration > Agent Criteria Groups. The Agent Criteria Group window appears in the work area.
3. Choose . The Agent Criteria Group Details pop-up window appears.



4. In Agent Criteria Group, enter the agent criteria group name.
5. In Short Description, enter the name of the agent criteria group.
6. In Long Description, enter a brief description of the agent criteria group.
7. Choose .

6.4.2 Modifying an Agent Criteria Group

To modify an agent criteria group:

1. From the menu bar, choose Applications > Platform. The Platform tree is displayed in the side panel.
2. From the Platform tree, choose System Administration > Agent Criteria Groups. The Agent Criteria Group window appears in the work area.
3. Select the Agent Criteria Group you want to modify and choose . The Agent Criteria Group Details pop-up window appears.
4. In Short Description, modify the name of the agent criteria group as needed.
5. In Long Description, modify the brief description of the agent criteria group as needed.
6. Choose .

6.4.3 Deleting an Agent Criteria Group

To delete an agent criteria group:

1. From the menu bar, choose Applications > Platform. The Platform tree is displayed in the side panel.
2. From the Platform tree, choose System Administration > Agent Criteria Groups. The Agent Criteria Group window appears in the work area.
3. Select the agent criteria group you want to delete and choose .

6.5 Defining Initial Context Factory Codes

You can configure additional initial context factory codes to be used to define the class providing an InitialContext implementation for your application server to enable remote Java clients to connect. These codes appear in the Initial Context Factory drop-down field used when configuring time-triggered transactions and services.

The default initial context factory codes and their class names are:

- WebSphere MQ - `com.ibm.websphere.naming.WsnInitialContextFactory`
- File - `com.sun.jndi.fscontext.RefFSContextFactory`
- WebLogic - `weblogic.jndi.WLInitialContextFactory`

You can use the Initial Context Factory Codes branch for:

- [Creating an Initial Context Factory Code](#)
- [Modifying an Initial Context Factory Code](#)
- [Deleting an Initial Context Factory Code](#)

6.5.1 Creating an Initial Context Factory Code

To create an initial context factory code:

1. From the tree in the application rules side panel, choose Nomenclature > Initial Context Factory Codes. The Initial Context Factory Codes window appears in the work area.
2. Choose . The Initial Context Factory Details pop-up window appears.

3. In Initial Context Factory, enter the class name to be associated with the code.

Important: The class name must be unique or an error is thrown.

4. In Short Description, enter the name of the initial context factory code as you want it to appear in the drop-down menus.
5. In Long Description, enter a brief description of the initial context factory code.
6. Choose .

6.5.2 Modifying an Initial Context Factory Code

To modify an initial context factory code:

1. From the tree in the application rules side panel, choose Nomenclature > Initial Context Factory Codes. The Initial Context Factory Codes window appears in the work area.
2. Select the code you want to modify and choose . The Initial Context Factory Details pop-up window appears.
3. In Short Description, enter the name of the initial context factory code as you want it to appear in the drop-down menus.

4. In Long Description, enter a brief description of the initial context factory code.
5. Choose .

6.5.3 Deleting an Initial Context Factory Code

To delete an initial context factory code:

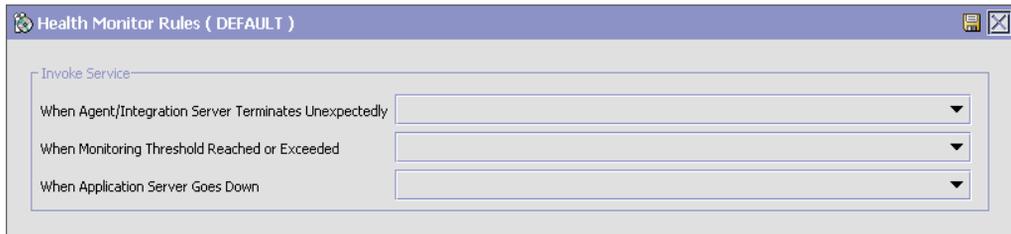
1. From the tree in the application rules side panel, choose Nomenclature > Initial Context Factory Codes. The Initial Context Factory Codes window appears in the work area.
2. Select the code you want to delete and choose .

6.6 Defining Health Monitor Rules

You can set up rules that need to be defined for monitoring the health of your Yantra 7x application.

To set up health monitor rules:

3. From the Platform tree, choose System Administration > Health Monitor Rules. The Health Monitor Rules window appears in the work area.
4. Enter information in the applicable fields. Refer to [Table 7-16](#) for field value descriptions. For more information about the health monitor agent or monitor thresholds, see the *Yantra 7x System Management Guide*.
5. Choose .



Health Monitor Rules (DEFAULT)

Invoke Service

When Agent/Integration Server Terminates Unexpectedly

When Monitoring Threshold Reached or Exceeded

When Application Server Goes Down

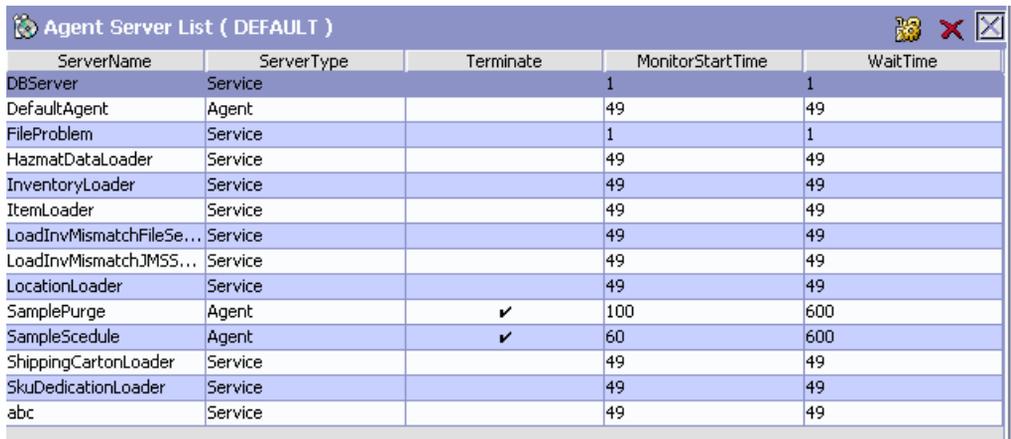
Table 6–5 Health Monitor Rules Window

Field	Description
Invoke Service	
When Agent/Integration Server Terminates Unexpectedly	Select a service to execute when an agent or integration server terminates unexpectedly.
When Monitoring Threshold Reached or Exceeded	Select a service to execute when a monitoring threshold for API response time, application server response time, agent pending tasks, or JMS queue number of messages is reached or exceeded for three consecutive health monitor persist intervals.
When Application Server Goes Down	Select a service to execute whenever an application server goes down.

6.7 Viewing the List of Configured Servers

You can view a list of configured servers for agents and services in the Agent Server List screen. To view the list of servers:

1. From the Platform tree in the application rules side panel, choose System Administration > Configured Servers. The Agent Server List window appears in the work area as shown below:



ServerName	ServerType	Terminate	MonitorStartTime	WaitTime
DBServer	Service		1	1
DefaultAgent	Agent		49	49
FileProblem	Service		1	1
HazmatDataLoader	Service		49	49
InventoryLoader	Service		49	49
ItemLoader	Service		49	49
LoadInvMismatchFileSe...	Service		49	49
LoadInvMismatchJMSS...	Service		49	49
LocationLoader	Service		49	49
SamplePurge	Agent	✓	100	600
SampleSchedule	Agent	✓	60	600
ShippingCartonLoader	Service		49	49
SkuDedicationLoader	Service		49	49
abc	Service		49	49

Table 6–6 Agent Server List

Field	Description
Server Name	The name of the agent server.
Server Type	The type of the server. For example, Agent and Service are valid server types.
Terminate	This option specifies whether the server is terminated when the task is completed.
Monitor Start Time (seconds)	Specifies the monitor start time. This is to ensure that the server does not terminate even before it has completed one successful execution.
Wait Time (seconds)	Specifies the idle wait time before terminating the server.

Note: A server can be deleted only if there are no services or agents configured to use it.

- The list of services or agents configured for this server can be viewed by selecting . Refer to [Section 6.7.1, "List of Sub Flows or Criteria ID Configured for Server"](#) to view the services or agent configured on the Server.
- The factory default agents provided in Yantra 7x does not have the "Terminate" option configured by default.

6.7.1 List of Sub Flows or Criteria ID Configured for Server

You can view the list of sub flows or criteria IDs configured for a server in the Flow List For Server screen. For more information on the field details see ["Adding a New Server"](#) in [Chapter 4, "Configuring Process Models"](#).

Configuring Units of Measure

Defining units of measure enables you to set up standard units of measure to associate with your items and locales. Defining units of measure ensures that each user sees the data in a familiar format.

You can use the Unit Of Measure branch for:

- [Defining Quantity Units of Measure](#)
- [Defining Service Quantity Units of Measure](#)
- [Defining Dimension Units of Measure](#)
- [Defining Volume Units of Measure](#)
- [Defining Weight Units of Measure](#)
- [Defining Time Units of Measure](#)

7.1 Defining Quantity Units of Measure

The Quantity branch enables you complete the following tasks:

- [Creating a Unit of Measure for Quantity](#)
- [Modifying a Unit of Measure for Quantity](#)
- [Creating a Unit of Measure Conversion Rate for Quantity](#)
- [Modifying a Unit of Measure Conversion Rate for Quantity](#)
- [Deleting a Unit of Measure Conversion Rate for Quantity](#)
- [Deleting a Unit of Measure for Quantity](#)

7.1.1 Creating a Unit of Measure for Quantity

To create a unit of measure for quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Quantity. The Quantity UOMs window appears in the work area.
2. Choose . The Unit of Measure Details window is displayed.
3. In UOM Code, enter the unit of measure.
4. In UOM Description, enter a brief description of the unit of measure.
5. Choose .

7.1.2 Modifying a Unit of Measure for Quantity

To modify a unit of measure for quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Quantity. The Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window is displayed.
3. In UOM Description, enter a brief description of the unit of measure.
4. Choose .

7.1.3 Creating a Unit of Measure Conversion Rate for Quantity

To create a unit of measure conversion for quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Quantity. The Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Choose . The UOM Conversion Details window is displayed.
4. From Conversion To, select the unit of measure you want to convert to with the conversion rate.

5. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
6. Choose .

7.1.4 Modifying a Unit of Measure Conversion Rate for Quantity

To modify a unit of measure conversion for quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Quantity. The Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window is displayed.
4. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
5. Choose .

7.1.5 Deleting a Unit of Measure Conversion Rate for Quantity

To delete a unit of measure conversion for quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Quantity. The Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose .

7.1.6 Deleting a Unit of Measure for Quantity

To delete a unit of measure for quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Quantity. The Quantity UOMs window appears in the work area.

2. Select the applicable unit of measure and choose .

7.2 Defining Service Quantity Units of Measure

The Service Quantity branch enables you complete the following tasks:

- [Creating a Unit of Measure for Service Quantity](#)
- [Modifying a Unit of Measure for Service Quantity](#)
- [Creating a Unit of Measure Conversion Rate for Service Quantity](#)
- [Modifying a Unit of Measure Conversion Rate for Service Quantity](#)
- [Deleting a Unit of Measure Conversion Rate for Service Quantity](#)
- [Deleting a Unit of Measure for Service Quantity](#)

7.2.1 Creating a Unit of Measure for Service Quantity

To create a unit of measure for service quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Service Quantity. The Service Quantity UOMs window appears in the work area.
2. Choose . The Unit of Measure Details window is displayed.
3. In UOM Code, enter the unit of measure.
4. In UOM Description, enter a brief description of the unit of measure.
5. Choose .

7.2.2 Modifying a Unit of Measure for Service Quantity

To modify a unit of measure for service quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Service Quantity. The Service Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window is displayed.
3. In UOM Description, enter a brief description of the unit of measure.
4. Choose .

7.2.3 Creating a Unit of Measure Conversion Rate for Service Quantity

To create a unit of measure conversion for service quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Service Quantity. The Service Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Choose . The UOM Conversion Details window is displayed.
4. From Conversion To, select the unit of measure you want to convert to with the conversion rate.
5. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
6. Choose .

7.2.4 Modifying a Unit of Measure Conversion Rate for Service Quantity

To modify a unit of measure conversion for service quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Service Quantity. The Service Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window is displayed.
4. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
5. Choose .

7.2.5 Deleting a Unit of Measure Conversion Rate for Service Quantity

To delete a unit of measure conversion for service quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Service Quantity. The Service Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose .

7.2.6 Deleting a Unit of Measure for Service Quantity

To delete a unit of measure for service quantity:

1. From the tree in the application rules side panel, choose Unit of Measure > Service Quantity. The Service Quantity UOMs window appears in the work area.
2. Select the applicable unit of measure and choose .

7.3 Defining Dimension Units of Measure

The Dimension branch enables you complete the following tasks:

- [Creating a Unit of Measure for Dimension](#)
- [Modifying a Unit of Measure for Dimension](#)
- [Creating a Unit of Measure Conversion Rate for Dimension](#)
- [Modifying a Unit of Measure Conversion Rate for Dimension](#)
- [Deleting a Unit of Measure Conversion Rate for Dimension](#)
- [Deleting a Unit of Measure for Dimension](#)

7.3.1 Creating a Unit of Measure for Dimension

To create a unit of measure for dimension:

1. From the tree in the application rules side panel, choose Unit of Measure > Dimension. The Dimension UOMs window appears in the work area.
2. Choose . The Unit of Measure Details window is displayed.
3. In UOM Code, enter the unit of measure.
4. In UOM Description, enter a brief description of the unit of measure.

5. From Volume UOM Code, select the volume unit of measure that corresponds to the volume measurement for this dimension. For example, choose cubic inches if the dimension unit of measure is inches.
6. Choose .

7.3.2 Modifying a Unit of Measure for Dimension

To modify a unit of measure for dimension:

1. From the tree in the application rules side panel, choose Unit of Measure > Dimension. The Dimension UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window is displayed.
3. In UOM Description, enter a brief description of the unit of measure.
4. Choose .

7.3.3 Creating a Unit of Measure Conversion Rate for Dimension

To create a unit of measure conversion for dimension:

1. From the tree in the application rules side panel, choose Unit of Measure > Dimension. The Dimension UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Choose . The UOM Conversion Details window is displayed.
4. From Conversion To, select the unit of measure you want to convert to with the conversion rate.
5. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
6. Choose .

7.3.4 Modifying a Unit of Measure Conversion Rate for Dimension

To modify a unit of measure conversion for dimension:

1. From the tree in the application rules side panel, choose Unit of Measure > Dimension. The Dimension UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window is displayed.
4. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
5. Choose .

7.3.5 Deleting a Unit of Measure Conversion Rate for Dimension

To delete a unit of measure conversion for dimension:

1. From the tree in the application rules side panel, choose Unit of Measure > Dimension. The Dimension UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose .

7.3.6 Deleting a Unit of Measure for Dimension

To delete a unit of measure for dimension:

1. From the tree in the application rules side panel, choose Unit of Measure > Dimension. The Dimension UOMs window appears in the work area.
2. Select the applicable unit of measure and choose .

7.4 Defining Volume Units of Measure

The Volume branch enables you complete the following tasks:

- [Creating a Unit of Measure for Volume](#)
- [Modifying a Unit of Measure for Volume](#)
- [Creating a Unit of Measure Conversion Rate for Volume](#)
- [Modifying a Unit of Measure Conversion Rate for Volume](#)
- [Deleting a Unit of Measure Conversion Rate for Volume](#)
- [Deleting a Unit of Measure for Volume](#)

7.4.1 Creating a Unit of Measure for Volume

To create a unit of measure for volume:

1. From the tree in the application rules side panel, choose Unit of Measure > Volume. The Volume UOMs window appears in the work area.
2. Choose . The Unit of Measure Details window is displayed.
3. In UOM Code, enter the unit of measure.
4. In UOM Description, enter a brief description of the unit of measure.
5. Choose .

7.4.2 Modifying a Unit of Measure for Volume

To modify a unit of measure for volume:

1. From the tree in the application rules side panel, choose Unit of Measure > Volume. The Volume UOMs window appears in the work area.
2. Select the applicable unit of measure to modify and choose . The Unit of Measure Details window is displayed.
3. In UOM Description, enter a brief description of the unit of measure.
4. Choose .

7.4.3 Creating a Unit of Measure Conversion Rate for Volume

To create a unit of measure conversion for volume:

1. From the tree in the application rules side panel, choose Unit of Measure > Volume. The Volume UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Choose . The UOM Conversion Details window is displayed.
4. From Conversion To, select the unit of measure you want to convert to with the conversion rate.
5. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
6. Choose .

7.4.4 Modifying a Unit of Measure Conversion Rate for Volume

To modify a unit of measure conversion for volume:

1. From the tree in the application rules side panel, choose Unit of Measure > Volume. The Volume UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window is displayed.
4. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
5. Choose .

7.4.5 Deleting a Unit of Measure Conversion Rate for Volume

To delete a unit of measure conversion for volume:

1. From the tree in the application rules side panel, choose Unit of Measure > Volume. The Volume UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose .

7.4.6 Deleting a Unit of Measure for Volume

To delete a unit of measure for volume:

1. From the tree in the application rules side panel, choose Unit of Measure > Volume. The Volume UOMs window appears in the work area.
2. Select the applicable unit of measure and choose .

7.5 Defining Weight Units of Measure

The Weight branch enables you complete the following tasks:

- [Creating a Unit of Measure for Weight](#)
- [Modifying a Unit of Measure for Weight](#)
- [Creating a Unit of Measure Conversion Rate for Weight](#)
- [Modifying a Unit of Measure Conversion Rate for Weight](#)
- [Deleting a Unit of Measure Conversion Rate for Weight](#)
- [Deleting a Unit of Measure for Weight](#)

7.5.1 Creating a Unit of Measure for Weight

To create a unit of measure for weight:

1. From the tree in the application rules side panel, choose Unit of Measure > Weight. The Weight UOMs window appears in the work area.
2. Choose . The Unit of Measure Details window is displayed.
3. In UOM Code, enter the unit of measure.
4. In UOM Description, enter a brief description of the unit of measure.

5. Choose .

7.5.2 Modifying a Unit of Measure for Weight

To modify a unit of measure for weight:

1. From the tree in the application rules side panel, choose Unit of Measure > Weight. The Weight UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window is displayed.
3. In UOM Description, enter a brief description of the unit of measure.
4. Choose .

7.5.3 Creating a Unit of Measure Conversion Rate for Weight

To create a unit of measure conversion for weight:

1. From the tree in the application rules side panel, choose Unit of Measure > Weight. The Weight UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Choose . The UOM Conversion Details window is displayed.
4. From Conversion To, select the unit of measure you want to convert to with the conversion rate.
5. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
6. Choose .

7.5.4 Modifying a Unit of Measure Conversion Rate for Weight

To modify a unit of measure conversion for weight:

1. From the tree in the application rules side panel, choose Unit of Measure > Weight. The Weight UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window is displayed.
4. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
5. Choose .

7.5.5 Deleting a Unit of Measure Conversion Rate for Weight

To delete a unit of measure conversion for weight:

1. From the tree in the application rules side panel, choose Unit of Measure > Weight. The Weight UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose .

7.5.6 Deleting a Unit of Measure for Weight

To delete a unit of measure for weight:

1. From the tree in the application rules side panel, choose Unit of Measure > Weight. The Weight UOMs window appears in the work area.
2. Select the applicable unit of measure and choose .

7.6 Defining Time Units of Measure

The Time branch enables you complete the following tasks:

- [Creating a Unit of Measure for Time](#)
- [Modifying a Unit of Measure for Time](#)

- [Creating a Unit of Measure Conversion Rate for Time](#)
- [Modifying a Unit of Measure Conversion Rate for Time](#)
- [Deleting a Unit of Measure Conversion Rate for Time](#)
- [Deleting a Unit of Measure for Time](#)

7.6.1 Creating a Unit of Measure for Time

To create a unit of measure for time:

1. From the tree in the application rules side panel, choose Unit of Measure > Time. The Time UOMs window appears in the work area.
2. Choose . The Unit of Measure Details window is displayed.
3. In UOM Code, enter the unit of measure.
4. In UOM Description, enter a brief description of the unit of measure.
5. Choose .

7.6.2 Modifying a Unit of Measure for Time

To modify a unit of measure for time:

1. From the tree in the application rules side panel, choose Unit of Measure > Time. The Time UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window is displayed.
3. In UOM Description, enter a brief description of the unit of measure.
4. Choose .

7.6.3 Creating a Unit of Measure Conversion Rate for Time

To create a unit of measure conversion for time:

1. From the tree in the application rules side panel, choose Unit of Measure > Time. The Time UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Choose . The UOM Conversion Details window is displayed.

4. From Conversion To, select the unit of measure you want to convert to with the conversion rate.
5. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
6. Choose .

7.6.4 Modifying a Unit of Measure Conversion Rate for Time

To modify a unit of measure conversion for time:

1. From the tree in the application rules side panel, choose Unit of Measure > Time. The Time UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window is displayed.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window is displayed.
4. In Conversion Rate, enter the conversion rate between Conversion From to Conversion To.
5. Choose .

7.6.5 Deleting a Unit of Measure Conversion Rate for Time

To delete a unit of measure conversion for time:

1. From the tree in the application rules side panel, choose Unit of Measure > Time. The Time UOMs window appears in the work area.
2. Select the applicable unit of measure and choose . The UOM Conversion Details window is displayed.
3. Select the applicable conversion rate and choose .

7.6.6 Deleting a Unit of Measure for Time

To delete a unit of measure for time:

1. From the tree in the application rules side panel, choose Unit of Measure > Time. The Time UOMs window appears in the work area.
2. Select the applicable unit of measure and choose .

Configuring Internationalization Rules

Internationalization rules are used to set up rules and common codes associated with making Yantra 7x functional for international use.

You can use the Internationalization branch for:

- [Defining Country Codes](#)
- [Defining Language Codes](#)
- [Defining Date Formats](#)
- [Defining Time Formats](#)
- [Defining Date/Time Formats](#)
- [Defining Currency Definitions](#)
- [Defining Currency Conversions](#)

8.1 Defining Country Codes

You can set up common codes for country codes used when setting up locales. This common code identifies the country that the locale is located in.

The following are examples of country codes:

- US (United States)
- FR (France)
- GB (United Kingdom)

You can use the Countries branch for:

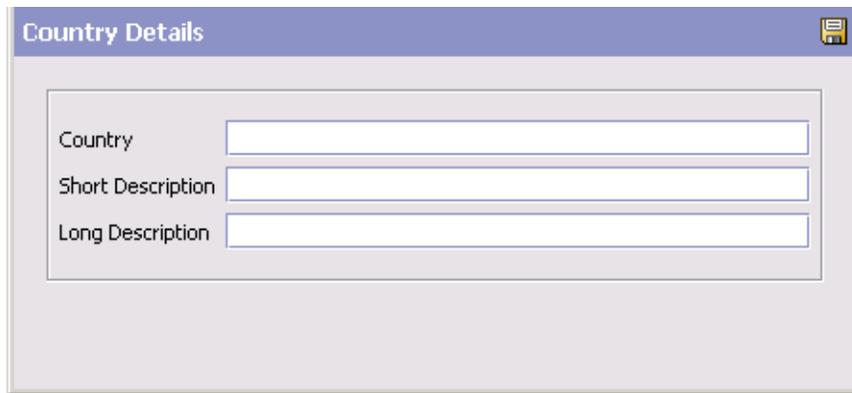
- [Creating a Country Code Definition](#)

- [Modifying a Country Code Definition](#)
- [Deleting a Country Code Definition](#)

8.1.1 Creating a Country Code Definition

To create a country code definition:

1. From the tree in the application rules side panel, choose Internationalization > Countries. The Country Codes window appears in the work area.
2. Choose . The Country Details pop-up window appears.



The image shows a dialog box titled "Country Details" with a save icon in the top right corner. Inside the dialog, there are three text input fields labeled "Country", "Short Description", and "Long Description".

3. In Country, enter a two character country code definition.

Note: The country code should be in accordance with the International Standard Organization (ISO) specifications (the two character code in the ISO 3166-1 document).

4. In Short Description, enter a brief description of the country code definition.
5. In Long Description, enter a more detailed description of the country code definition.
6. Choose .

8.1.2 Modifying a Country Code Definition

To modify a country code definition:

1. From the tree in the application rules side panel, choose Internationalization > Countries. The Country Codes window appears in the work area.
2. Select the applicable country code definition and choose . The Country Details pop-up window appears.
3. In Short Description, enter a brief description of the country code definition.
4. In Long Description, enter a more detailed description of the country code definition.
5. Choose .

8.1.3 Deleting a Country Code Definition

To delete a country code definition:

1. From the tree in the application rules side panel, choose Internationalization > Countries. The Country Codes window appears in the work area.
2. Select the applicable country code definition and choose .

8.2 Defining Language Codes

You can set up common codes for language definitions used when setting up locales. This common code identifies the language used in the locale. You can create, modify, and delete language definitions.

The following is Yantra 7x default language definition:

- EN (English)

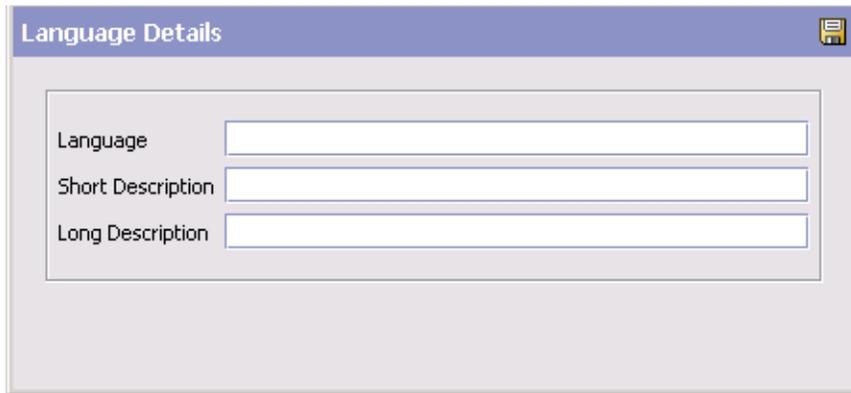
You can use the Languages branch for:

- [Creating a Language Definition](#)
- [Modifying a Language Definition](#)
- [Deleting a Language Definition](#)

8.2.1 Creating a Language Definition

To create a language definition:

1. From the tree in the application rules side panel, choose Internationalization > Languages. The Language Codes window appears in the work area.
2. Choose . The Language Definition Details pop-up window appears.



The screenshot shows a dialog box titled "Language Details". It has a blue header bar with the title and a close button on the right. The main area is light gray and contains three text input fields stacked vertically. The first field is labeled "Language", the second "Short Description", and the third "Long Description".

3. In Language Definition, enter a two character language definition.

Note: The language definition should be in accordance with the International Standard Organization (ISO) specifications (the two character code from the ISO 639 document).

4. In Short Description, enter a brief description of the language definition.
5. In Long Description, enter a more detailed description of the language definition.
6. Choose .

8.2.2 Modifying a Language Definition

To modify a language definition:

1. From the tree in the application rules side panel, choose Internationalization > Languages. The Language Codes window appears in the work area.
2. Select the applicable language definition and choose . The Language Definition Details pop-up window appears.
3. In Short Description, enter a brief description of the language definition.
4. In Long Description, enter a more detailed description of the language definition.
5. Choose .

8.2.3 Deleting a Language Definition

To delete a language definition:

1. From the tree in the application rules side panel, choose Internationalization > Languages. The Language Codes window appears in the work area.
2. Select the applicable language definition and choose .

8.3 Defining Date Formats

You can set up common code formats for date formats used when setting up locales. This common code format identifies how dates are entered at a locale. You can create, modify, and delete date formats.

The following are examples of date formats:

- MM/dd/yyyy
- dd/MM/yyyy

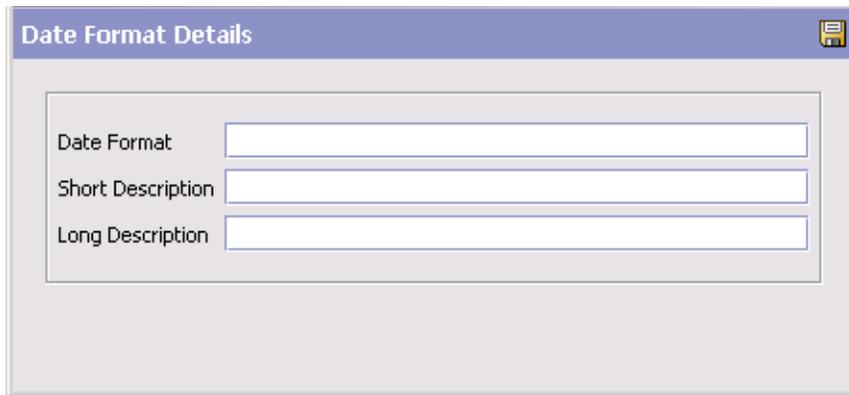
You can use the Date Formats branch for:

- [Creating a Date Format](#)
- [Modifying a Date Format](#)
- [Deleting a Date Format](#)

8.3.1 Creating a Date Format

To create a date format:

1. From the tree in the application rules side panel, choose Internationalization > Date Formats. The Date Formats window appears in the work area.
2. Choose . The Date Format Details pop-up window appears.



The image shows a dialog box titled "Date Format Details". It has a blue header bar with the title and a save icon. Below the header is a light gray area containing three text input fields. The first field is labeled "Date Format", the second is labeled "Short Description", and the third is labeled "Long Description".

3. In Date Format, enter the date format using 'M' to stand for month, 'd' to stand for day, and 'y' to stand for year.
4. In Short Description, enter a brief description of the date format.
5. In Long Description, enter a more detailed description of the date format.
6. Choose .

8.3.2 Modifying a Date Format

To modify a date format:

1. From the tree in the application rules side panel, choose Internationalization > Date Formats. The Date Formats window appears in the work area.
2. Select the applicable date format and choose . The Date Format Details pop-up window appears.
3. In Short Description, enter a brief description of the date format.

4. In Long Description, enter a more detailed description of the date format.
5. Choose .

8.3.3 Deleting a Date Format

To delete a date format:

1. From the tree in the application rules side panel, choose Internationalization > Date Formats. The Date Formats window appears in the work area.
2. Select the applicable date format and choose .

8.4 Defining Time Formats

You can set up common code formats for time formats used when setting up locales. This common code format identifies how times are entered at a locale. You can create, modify, and delete time formats.

Note: Yantra 7x uses Java time/date conventions. For example, if you enter hh for hour you are indicating you want to use a 12 hour clock. If you enter HH for hour you are indicating you want to use a 24 hour clock. However, the application only supports the 24 hour clock.

You can use the Time Formats branch for:

- [Creating a Time Format](#)
- [Modifying a Time Format](#)
- [Deleting a Time Format](#)

8.4.1 Creating a Time Format

To create a time format:

1. From the tree in the application rules side panel, choose Internationalization > Time Formats. The Time Formats window appears in the work area.
2. Choose . The Time Format Details pop-up window appears.



The screenshot shows a dialog box titled "Time Format Details". It has a blue header bar with the title and a save icon. The main area is light gray and contains three text input fields stacked vertically. The first field is labeled "Time Format", the second "Short Description", and the third "Long Description".

3. In Time Format, enter the date format using 'H' to stand for hour, 'm' to stand for minute, and 's' to stand for second.
4. In Short Description, enter a brief description of the time format.
5. In Long Description, enter a more detailed description of the time format.
6. Choose .

8.4.2 Modifying a Time Format

To modify a time format:

1. From the tree in the application rules side panel, choose Internationalization > Time Formats. The Time Formats window appears in the work area.
2. Select the applicable time format and choose . The Time Format Details pop-up window appears.
3. In Short Description, enter a brief description of the time format.
4. In Long Description, enter a more detailed description of the time format.
5. Choose .

8.4.3 Deleting a Time Format

To delete a time format:

1. From the tree in the application rules side panel, choose Internationalization > Time Formats. The Time Formats window appears in the work area.
2. Select the applicable time format and choose .

8.5 Defining Date/Time Formats

You can set up common code formats for date/time formats used when setting up locales. This common code format identifies how dates with time are entered at a locale. You can create, modify, and delete date/time formats.

Note: Yantra 7x uses Java time/date conventions. For example, if you enter hh for hour you are indicating you want to use a 12 hour clock. If you enter HH for hour you are indicating you want to use a 24 hour clock. However, the application only supports the 24 hour clock.

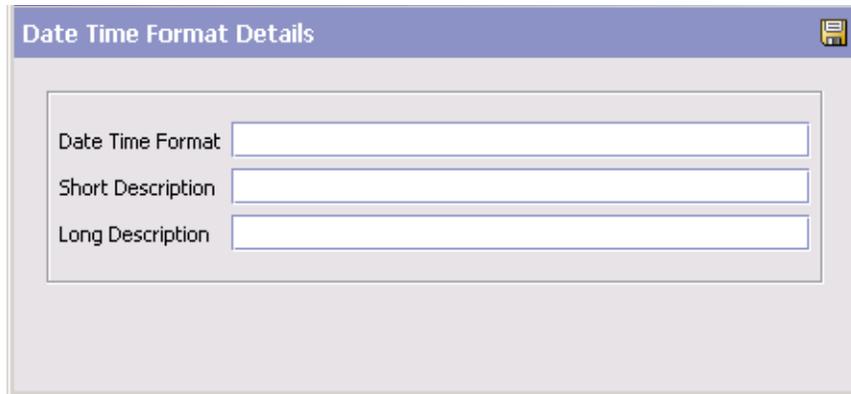
You can use the Date Time Formats branch for:

- [Creating a Date/Time Format](#)
- [Modifying a Date/Time Format](#)
- [Deleting a Date/Time Format](#)

8.5.1 Creating a Date/Time Format

To create a date/time format:

1. From the tree in the application rules side panel, choose Internationalization > Date Time Formats. The Time Formats window appears in the work area.
2. Choose . The Date/Time Format Details pop-up window appears.



The screenshot shows a dialog box titled "Date Time Format Details". It contains three text input fields:

- Date Time Format
- Short Description
- Long Description

3. In Date/Time Format, enter the date/time format using 'M' to stand for month, 'd' to stand for day, 'y' to stand for year, 'h' to stand for hour, 'm' to stand for minute, and 's' to stand for second.
4. In Short Description, enter a brief description of the date/time format.
5. In Long Description, enter a more detailed description of the date/time format.
6. Choose .

8.5.2 Modifying a Date/Time Format

To modify a date/time format:

1. From the tree in the application rules side panel, choose Internationalization > Date Time Formats. The Time Formats window appears in the work area.
2. Select the applicable date/time format and choose . The Date/Time Format Details pop-up window appears.
3. In Short Description, enter a brief description of the date/time format.
4. In Long Description, enter a more detailed description of the date/time format.
5. Choose .

8.5.3 Deleting a Date/Time Format

To delete a date/time format:

1. From the tree in the application rules side panel, choose Internationalization > Date Time Formats. The Time Formats window appears in the work area.
2. Select the applicable date/time format and choose .

8.6 Defining Currency Definitions

Currency Definitions define each currency's symbols and indicate Euro currency membership and expiration date, if applicable. You can also set rules for an order's currency conversion and euro conversion.

The Euro currency is part of the plan to convert all of the European nations to one defined currency. The following countries' use the euro as their currency as of August 2003:

- Austria
- Belgium
- Finland
- France
- Germany
- Ireland
- Italy
- Luxembourg
- The Netherlands
- Portugal
- Spain

You can use the Currency Definitions branch for:

- [Creating a Currency Definition](#)
- [Modifying a Currency Definition](#)
- [Deleting a Currency Definition](#)

8.6.1 Creating a Currency Definition

To create currency definitions:

1. From the tree in the application rules side panel, choose Internationalization > Currency Definitions. The Currency Definition window appears in the work area.
2. Choose . The Currency Details pop-up window is displayed.
3. Enter information in the applicable fields. Refer to [Table 7–17](#) for field value descriptions.
4. Choose .

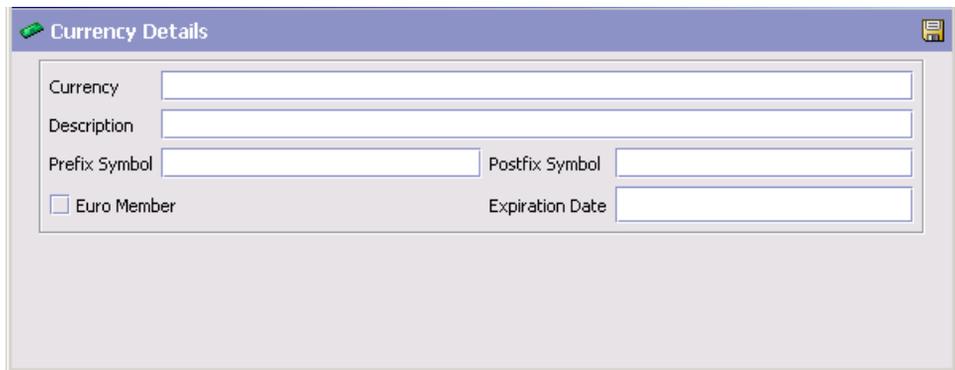


Table 8–1 *Currency Details Pop-Up Window*

Field	Description
Currency	Enter the name of the currency.
Description	Enter a brief description of the currency.
Prefix Symbol	Enter the symbol that precedes the currency amount (if applicable).
Postfix Symbol	Enter the symbol that follows the currency amount (if applicable).

Table 8–1 Currency Details Pop-Up Window

Field	Description
Euro Member	Select the check box if the currency is a Euro Member Currency.
Expiration Date	If Euro Member is selected, enter the date that the currency is set to expire. For example, if you are creating a currency definition for one of the original 11 Euro Members, enter `01/01/2002'.

8.6.2 Modifying a Currency Definition

To modify currency definitions:

1. From the tree in the application rules side panel, choose Internationalization > Currency Definitions. The Currency Definition window appears in the work area.
2. Select the applicable currency definition and choose . The Currency Details pop-up window is displayed.
3. Enter information in the applicable fields. Refer to [Table 7–18](#) for field value descriptions.
4. Choose .

8.6.3 Deleting a Currency Definition

To delete currency definitions:

1. From the tree in the application rules side panel, choose Internationalization > Currency Definitions. The Currency Definition window appears in the work area.
2. Select the applicable currency definition and choose .

8.6.3.1 Defining the Default Reporting Conversion Rate and Euro Currency Code

To define the default reporting conversion rate and euro currency code:

1. From the tree in the application rules side panel, choose Internationalization > Currency Definitions. The Currency Definition window appears in the work area.

2. Check Default Reporting Conversion Rate to default the reporting conversion rate on the order at the time it is created.
3. From Euro Currency Code, select the currency to represent the euro currency in conversions, if applicable.

8.7 Defining Currency Conversions

Currency Conversion enables you to set up exchange rates from one currency to another.

Exchange rates are used to translate between currencies used by organizations as defined by their locale. The exchange rate is stored as part of the order document type when it is created. The stored exchange rate can be reassessed, based on fluctuating currency markets or any time the price of an order changes, such as when you cancel a line, add quantity, or add a charge.

You can use the Currency Conversions branch for:

- [Creating a Currency Conversion](#)
- [Modifying a Currency Conversion](#)
- [Deleting a Currency Conversion](#)

8.7.1 Creating a Currency Conversion

To create a currency conversion:

1. From the tree in the application rules side panel, choose Internationalization > Currency Conversions. The Currency Conversion window appears in the work area.
2. Choose . The Currency Conversion Details window is displayed.
3. Enter information in the applicable fields. Refer to [Table 7–18](#) for field value descriptions.
4. Choose .

Currency Conversion Details

Primary Information

Base Currency

Effective From

Effective To

Table 8–2 *Currency Conversion Details*

Field	Description
Base Currency	Select the currency you want to convert to and from.
Effective From	Enter the beginning day on which the exchange rate is valid.
Effective To	Enter the last day on which the exchange rate is valid.
Other Currency	Select a currency to use for conversion with the base currency.
Conversion Rate From Base	Enter the conversion rate from the Base Currency to the Other Currency.
Conversion Rate To Base	Enter the conversion rate from the Other Currency to the Base Currency.

8.7.2 Modifying a Currency Conversion

To modify a currency conversion:

1. From the tree in the application rules side panel, choose Internationalization > Currency Conversions. The Currency Conversion window appears in the work area.
2. Select the applicable currency conversion and choose . The Currency Conversion Details window is displayed.
3. Enter information in the applicable fields. Refer to [Table 7–18](#) for field value descriptions.
4. Choose .

8.7.3 Deleting a Currency Conversion

To delete a currency conversion:

1. From the tree in the application rules side panel, choose Internationalization > Currency Conversions. The Currency Conversion window appears in the work area.
2. Select the applicable currency conversion and choose .

Configuring Presentation Components

The Yantra 7x Presentation Framework provides an interface that enables you to customize the graphical user interface.

You can use the Presentation branch for:

- [Defining Locales](#)
- [Defining Menus](#)
- [Defining Resources](#)
- [Defining Themes](#)
- [Defining Custom Common Code Types](#)
- [Defining Custom Common Code Values](#)
- [Defining Custom Error Codes](#)

9.1 Defining Locales

You can set up locales and associate them with multiple organizations and users within the Hub. Locales are only established by the Hub. A **locale** defines a set of standards that enable people within a geographic area to communicate and conduct business transactions in a unambiguous way. For example, the locale defines the unique time zone, language, date/time format, currency, and units of measurement for a specific geographical area.

Locale standards enable international organizations to interact globally and ensure that UOM, currency conversions, and time zones calculations are taken into consideration during a transaction. For example, the following locale definitions for organizations in San Francisco and Tokyo enable them to transact business with each other:

Example 9–1 San Francisco, United States

- Locale - California
- Country - United States
- Language - English
- Date/Time - DD/MM/YYYY; HH:MM:SS
- Time Zone - GMT-8:00
- Currency - Dollar
- Unit of Measure - Inch/Pound

Example 9–2 Tokyo, Japan

- Locale - Tokyo
- Country - Japan
- Language - Japanese
- Date/Time - DD/MM/YYYY; HH:MM:SS
- Time Zone - GMT+9:00
- Currency - Yen
- Unit of Measure - Meter/Kilogram

You can use the Locales branch for:

- [Creating a Locale](#)
- [Modifying a Locale](#)
- [Deleting a Locale](#)

9.1.1 Creating a Locale

To create a locale:

1. From the tree in the application rules side panel, choose Presentation > Locales. The Locale window appears in the work area.
2. Choose . The Locale Details window is displayed.
3. Enter information in the applicable fields. Refer to [Table 9–1](#) for assistance.

4. Choose .

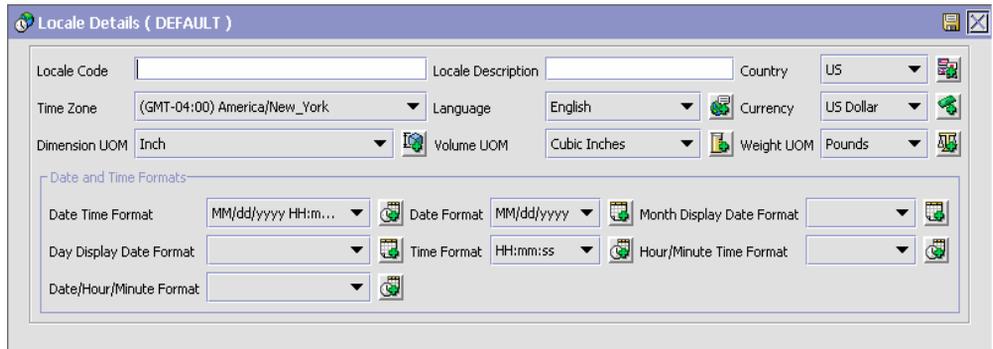


Table 9–1 *Locale Details Window*

Field	Description
Locale Code	Enter the name of the locale as you want it to appear throughout Yantra 7x. Cannot contain spaces. The recommended syntax is to join the entries for Country and Language with an underscore. For example, use fr_CA to represent <i>Canadian French</i> .
Locale Description	Enter a brief description of the locale.
Country	Select the country. Using the two-character upper-case ISO-3166 code is recommended. For example, enter CA for <i>Canada</i> .
Time Zone	Select the time zone of the locale.
Language	Select the language spoken in the locale. Using the two-character lower-case ISO-639 code is recommended. For example, enter fr for <i>French</i> .
Currency	Select the currency used in the locale.
Dimension UOM	Select the unit of measure the locale uses for dimension.
Volume UOM	Select the unit of measure the locale uses for volume.
Weight UOM	Select the unit of measure the locale uses for weight.
Date and Time Formats	

Table 9–1 *Locale Details Window*

Field	Description
Date Time Format	Select the date/time format used in the locale.
Date Format	Select the date format used in the locale.
Month Display Date Format	Select the date format that should be used when a month is displayed.
Day Display Date Format	Select the date format that should be used when a day of week is displayed.
Time Format	Select the time format used in the locale.
Hour/Minute Time Format	Select the time format that should be used when a time consisting of an hour and minutes is displayed.
Date/Hour/Minute Format	Select the date/time format that should be used when a that should be used when a date and time (consisting of an hour and minutes) is displayed.

9.1.2 Modifying a Locale

To modify a locale:

1. From the tree in the application rules side panel, choose Presentation > Locales. The Locale window appears in the work area.
2. Select the applicable locale and choose . The Locale Details window is displayed.
3. Enter information in the applicable fields. Refer to [Table 9–1](#) for assistance.
4. Choose .

9.1.3 Deleting a Locale

To delete a locale:

1. From the tree in the application rules side panel, choose Presentation > Locales. The Locale window appears in the work area.
2. Select the applicable locale and choose .

9.2 Defining Menus

You can define the menu (and screen structure) a user sees. Menu configuration contains the standard Yantra 7x resources as well as the ones that you define when configuring resources.

All menus are grouped into a menu group. The default menu group contains the standard menu configuration of the Yantra 7x Application Consoles.

No other applications can be added.

This default group is linked to the default Administrator user. When creating your own users, you can reuse this menu group or create a completely new menu group. Your own custom menus may contain different menu items.

You can use the Menu branch for:

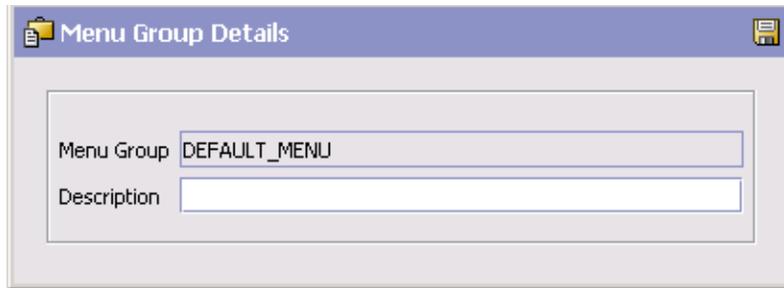
- [Saving a Menu Group as a New Menu Grouping](#)
- [Modifying Application Menu Details](#)
- [Defining Menu Items](#)
- [Modifying a Menu Item](#)
- [Deleting a Menu Item](#)

9.2.1 Saving a Menu Group as a New Menu Grouping

You can save an existing menu group as a new customizable menu group.

To save a menu group as a new menu grouping:

1. From the tree in the application rules side panel, choose Presentation > Menu. The Menu Hierarchy window appears in the work area.
2. Select the applicable menu group and choose . The Menu Group Details pop-up window is displayed



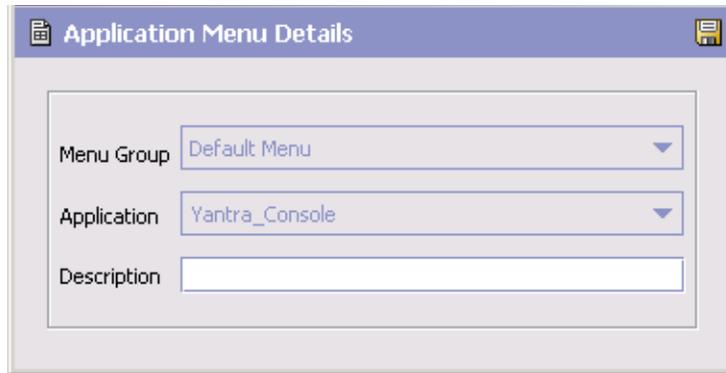
3. In Menu Group, enter the menu group name using the resource key used in the resource bundle.
4. In Description, enter the name of the menu group as you want it to appear in the menu hierarchy interface.
5. Choose .

9.2.2 Modifying Application Menu Details

You can modify the name of the application menu as it appears in the menu hierarchy.

To modify the Application Menu details:

1. From the tree in the application rules side panel, choose Presentation > Menu. The Menu Hierarchy window appears in the work area.
2. Expand the applicable menu group branch.
3. Select the applicable application menu and choose . The Application Menu Details pop-up window is displayed



4. In Description, enter the name of the application menu as you want it to appear in the menu hierarchy interface.
5. Choose .

9.2.3 Defining Menu Items

The menu hierarchy contains a list of all possible menu items you can provide to a user. You can create, modify, and delete menu items.

9.2.3.1 Creating a Menu Item

To create a menu item:

1. From the tree in the application rules side panel, choose Presentation > Menu. The Menu Hierarchy window appears in the work area.
2. Expand the applicable menu group branch.
3. Expand the applicable application menu.
4. Choose . The Menu Item Details pop-up window is displayed.
5. Enter information in the applicable fields. Refer to [Table 9–2](#) the [Menu Item Details Pop-Up Window](#) table for field value descriptions.
6. Choose .

The screenshot shows a window titled "Menu Item Details" with a save icon in the top right corner. The window contains the following fields:

- Menu Id:
- Description:
- Icon:
- Menu Sequence:
- Resource ID: (with a dropdown arrow and a small icon)

Table 9–2 Menu Item Details Pop-Up Window

Control	Description
Menu ID	Enter a value identifier for this menu option.
Description	Enter a value to correlate with the on-screen literals stored in the resource bundles.
Icon	<p>Enter a relative path to any images associated with the menu item. The image is picked up from an image .jar file, so specify these images as follows:</p> <ul style="list-style-type: none"> The default icons are located in the yantraiconsbe.jar file. Inside the .jar file, many of the icons are located in the console/icons directory. <p>The image is shown before the text in the UI.</p>

Table 9–2 Menu Item Details Pop-Up Window

Control	Description
Menu Sequence	Enter the menu sequence of the menu item. The sequence number is used to order the way menus appear at one level. By changing the menu sequence, you can switch the order.
Resource ID	<p>Enter the resource ID of the menu item.</p> <p>When a menu is tagged to an entity's resource ID in by default it means that when the user clicks on that menu on the screen it takes you to the first Search view available under that entity in Resource Configurator.</p> <p>The resource ID contains details about the screen that needs to be invoked when the menu item is selected. The results on the search window for this field shows all resources that are either detail view resources or entity resources.</p> <p>Note: If a resource of type Detail View is selected, the detail view is invoked when the menu is selected.</p> <p>If a resource of type entity is selected, the default search view of that entity is invoked.</p>

9.2.4 Modifying a Menu Item

To modify a menu item:

1. From the tree in the application rules side panel, choose Presentation > Menu. The Menu Hierarchy window appears in the work area.
2. Expand the applicable menu group branch.
3. Expand the applicable application menu.
4. Select the applicable menu item and choose . The Menu Item Details pop-up window is displayed.
5. Enter information in the applicable fields. Refer to [Table 9–2](#) the [Menu Item Details Pop-Up Window](#) table for field value descriptions.
6. Choose .

9.2.5 Deleting a Menu Item

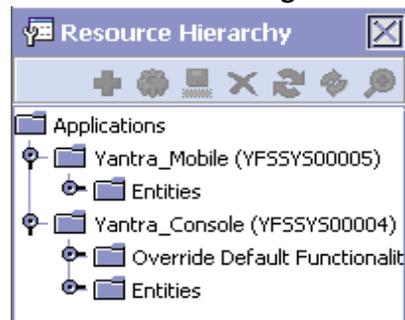
To delete a menu item:

1. From the tree in the application rules side panel, choose Presentation > Menu. The Menu Hierarchy window appears in the work area.
2. Expand the applicable menu group branch.
3. Expand the applicable application menu.
4. Select the applicable menu item and choose .

9.3 Defining Resources

A resource is a self-contained unit that controls the display of on-screen information and the execution of program logic associated with the display. For example, screens and APIs are types of resources.

Figure 9–1 Resource Configurator Tree



9.3.1 Yantra 7x Resources

All Yantra 7x resources have a set of primary properties (common to all resource types) and a set of unique properties (specific to a particular type of resource). Common primary properties are characteristics all resources have in common. For example, all resources have a Resource ID. These resources are used to define screens within the Yantra 7x Application Consoles.

For an explanation of all primary resource information, see [Table 9–3, "Primary Information of the Resource Details"](#).

In addition to primary information, each type of resource has unique characteristics. For information on each resource type, see the list of tables within this chapter.

Table 9–3 Primary Information of the Resource Details

Field	Description
Resource ID	<p>Unique identifier for each resource. Each type of resource has its own Resource ID syntax convention:</p> <ul style="list-style-type: none"> • API - <Parent Resource ID>AP<one up sequence number>. For example, the Order Console uses the <code>getOrderList()</code> API, whose Resource ID is <i>orderAP2</i>. • Detail View - Y<two-character module code> D<sequence number>. For example the Order Management has a detail view Resource ID <i>YOMD010</i>. • Entity- Free form. For example, <i>Order</i>. • Icon - <Parent Resource ID>C<two-digit sequence number>. For example, <i>YOMD010D1C02</i>. • Inner Panel - <Parent Resource ID>I<two-digit sequence number>. For example, <i>YOMD010I01</i>. • Link - <Parent Resource ID>L<two-digit sequence number>. For example, <i>YOML010</i>. • List View - Y<two-character module code> L<sequence number>. For example, <i>YOML010</i>. • Action - <Parent Resource ID>A<two-digit sequence number>. For example, <i>YOMD010I02A01</i>. • Related Entity - <Parent Entity ID>S<sequence number>. For example, <i>OrderS01</i>. • Search View - Y<two-character module code>S<sequence number>. For example, <i>YOMS010</i>.
Description	<p>Literal value displayed as a label on the target screen and within the Resource Configurator. This value is stored within the resource bundle.</p>

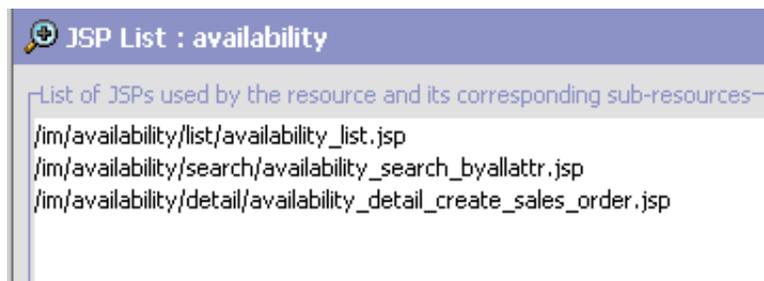
Table 9–3 Primary Information of the Resource Details

Field	Description
URL	For future use.
Resource Type	<p>Sets of characteristics that define a set of features on the user interface. For Application Consoles resources, see the following:</p> <ul style="list-style-type: none"> • Application Consoles Entity on page 305 • Application Consoles Related Entity on page 306 • Application Consoles Search View on page 307 • Application Consoles Detail View on page 308 • Application Consoles List View on page 309 • Application Consoles API on page 310 • Application Consoles Inner Panel on page 313 • Application Consoles Action on page 314 • Application Consoles Link on page 316 • Application Consoles Icon on page 317
Resource Sequence	<p>The order in which screen-related literals and logic are displayed or executed. Resource sequencing is valid only in the context of sibling resources.</p> <ul style="list-style-type: none"> • Related Entity - Order in which the entity's popup shows up in the search panel. • Search View - Order in which search views are listed in the search view popup menu. • Detail View - Order in which detail views are listed in the detail view popup menu. • List View - Order in which the list views are listed in the list view popup menu. • API - Order in which APIs are invoked. The sequencing is particularly important in the case of APIs, since the output of one API may be used as input to another. • Action - Order in which actions are displayed in the UI. • Icon - Order in which icon links to other views are displayed in the UI.

9.3.2 Application Consoles Entity

The Application Consoles Entity Resource Details dialog box enables you set the Document Type of an entity. An entity is an associated group of UI displays and program logic that pertain to a specific business entity, such as an order entity.

You can view the list of JSPs used by the resource and its corresponding sub-resources and the list of users with permissions to access the resource by performing a right-click on the entity and selecting JSP List or User List or by choosing  from the menu panel.



LoginId	UsergroupId
Adm1	SYSTEM
CSR1	SYSTEM
yantra	SYSTEM

Each entity has a default search view, list view, and detail view. A default view is determined by the sequencing of these views within the Resource Configurator. For example, if the Order entity has four search views, the default search view is determined as the one with the lowest resource sequence number among the four search views.

Under each entity resource, you can configure one detail API and one list API. The detail API configured is automatically called when a user navigates to the detail view of that entity. The list API is called when a user navigates to the list view of that entity.

You can prevent this default API from being called for a specific view by selecting the Ignore Default API parameter in the resource configuration screen.

Caution: While you *can* create any number of API Resources under an Entity, you should create **only one list** and **one detail API**.

Table 9–4 Entity Resource Dialog Box

Control	Description
Document Type	<p>When specified, this is passed as input to any APIs configured within this entity (including all sub-resources).</p> <p>Document Type is passed at the root node with the <code>DocumentType</code> attribute name. This is also used to pick up literals from the resource bundle when the <code>i18n</code> taglib is used in all the JSPs configured for this entity (and all sub-resources). Specifically, the application looks for an entry with <code><DocumentType>_<ResourceKey></code>. If that is not found, the entry for <code><ResourceKey></code> is picked up.</p>

9.3.3 Application Consoles Related Entity

The Application Consoles Related Entity Resource Details dialog box enables you to specify which entities appear as options on the drop-down menu of an entity on a search view.

Figure 9–2 Related Entities on a Drop-Down Menu



Table 9–5 Console Related Entity Resource Dialog Box

Control	Description
Related Entity	<p>The Resource ID of the related entity. Related Entities show up in search screens as other entities by which you can search. For example, when you choose Order from Menu, a search screen displays. In the drop-down menu on the top left corner of the search screen, you can see Order Lines and Order Releases. These are configured as related entities of Order entity.</p> <p>After you choose Order Line, your current entity becomes Order Line. Now, the drop-down menu shows Order and Order Releases as related entities, because they are explicitly defined as related entities for the entity Order Line.</p>

Permissions cannot be maintained for Related Entity resources. This means that all users are able to see the Related Entities in the entity drop-down of a search view.

9.3.4 Application Consoles Search View

The Console's Search View Resource Details dialog box enables you to specify details that enable you to customize a search view.

Table 9–6 Search View Resource Dialog Box

Control	Description
Java Server Page	<p>The JSP file name with full path relative to the <YFS_HOME>/webpages directory. Your custom resources typically points to a file within the /extn directory. For example, if you are changing the order search by status search view, specify the /extn/om/order/search/order_search_bystatus.jsp file.</p> <p>As you populate the /extn directory with custom JSP files, mimic the structure of the directory containing the standard JSP files.</p>
Output Name Space	<p>Namespace of the XML bindings of the search criteria input fields on the search JSP. The inputs that have this namespace are sent to the List API of the entity.</p>

Table 9–6 Search View Resource Dialog Box

Control	Description
Height	N/A
Width	N/A
Ignore Default API	N/A
Hide Maximum Records Field	Choose this to prevent the Max Records field from displaying.
View Group ID	Within a specific search view of an entity, you can switch to any other detail view having the same View Group ID as the current search view.
Input	N/A
Template	N/A

9.3.5 Application Consoles Detail View

The Application Consoles Detail View Resource Details dialog box enables you to customize a detail view.

Table 9–7 Detail View Resource Dialog Box

Control	Description
Java Server Page	<p>In the case of detail views, this field is used as the JSP for the anchor page. Typically, this JSP is used to simply include other inner panels and lay them out in the manner desired.</p> <p>If this field is not filled in, and the detail view has multiple inner panels, they are laid out one below the other, stretching all the way across horizontally.</p> <p>Specify the full path relative to the <YFS_HOME>/webpages directory. Your custom resources typically points to a file within the /extn directory. For example, if you are changing the anchor page of the order detail default view, your JSP might be /extn/om/order/detail/order_detail_anchor.jsp.</p> <p>As you populate the /extn directory with custom JSP files, mimic the structure of the directory containing the standard JSPs.</p>
Output Name Space	N/A

Table 9–7 Detail View Resource Dialog Box

Control	Description
Height	When this view appears in a popup screen, this is the height of the popup window.
Width	When this view appears in a popup screen, this is the width of the popup window.
Ignore Default API	<p>When this option is set, any detail API that might be configured for the current entity is not invoked when the current detail view is displayed.</p> <p>Typically, an entity has a standard detail API and several views that need different parts of the same detail API output. So, typically, you can simply define a detail API at the entity level and expect that the default detail API is called whenever a detail view for that entity is shown. However, some of the views of that entity may be special and may require a different API to be called. In such cases, this field can be checked and the alternate API can be configured for any one of the inner panels of the special detail view.</p>
Hide Navigation Panel	When this option is set, the Next and Previous navigation buttons do not show up on this detail view.
Hide Title Bar	When this option is set, the entire title bar does not show up for this detail view. Note that the title bar includes the screen title, Save, Help, and Close buttons.
View Group ID	Within a specific detail view of an entity, you can switch to any other detail view having the same View Group ID as the current detail view.
Input	N/A
Template	N/A
Redirector View	When this is set the detail view will be used only to redirect to another detail view. The JSP for this view should not contain any HTML intended for display. It should only contain conditional JSP statements that eventually call the <code>goToDetailView</code> JSP method to navigate to the final view.

9.3.6 Application Consoles List View

The Application Consoles List View Resource Details dialog box enables you to customize a list view.

Table 9–8 List View Resource Dialog Box

Control	Description
Java Server Page	The JSP file name with full path relative to the <YFS_HOME>/webpages directory. Your custom resources typically point to a file within the /extn directory. For example, if you are changing the order list concise view, specify the /extn/om/order/list/order_list_concise.jsp file. As you populate the /extn directory with custom JSP files, mimic the structure of the directory containing the standard JSP files.
Output Name Space	N/A
Height	N/A
Width	N/A
Ignore Default API checkbox	N/A
Supports Direct List to Detail Navigations When One Record Returned	When this is set and the related search results in one record, the list screen is bypassed and the details are displayed.
View Group ID	Within a specific list view of an entity, you can switch to any other detail view having the same View Group ID as the current list view.
Input	N/A
Template	When the default List API is invoked for a view, this is the output template that is passed to the API.

9.3.7 Application Consoles API

The Application Consoles API Resource Details dialog box enables you to specify whether or not to call an API and how that API is called.

Permissions cannot be maintained for API resources. An API resource can call a Yantra 7x standard API or invoke a service that has been configured.

Caution: Although you can create any number of API resources under an Entity, you *should* create **only one list** and **one detail API**.

Table 9–9 API Resource Dialog Box

Control	Description
Invoke a Service	Specifies that a service is invoked in the UI for this resource.
Service Name	Applies only to services. Enabled through the Invoke a Service radio button. Provides a drop-down list of available services that have been previously configured in the Services Configurator.
Templates	Applies only to services. Enabled through the Invoke a Service radio button. Provides a way for you to add, modify, and delete a list of templates. You can enter an API Name and a Template for each row in the table. This way you can enter templates for all APIs that are called within the service. Important: note that one service should not execute the same API twice because you cannot configure multiple template elements for the same API.
Invoke an API	Specifies that a Yantra 7x standard API is invoked in the UI for this resource. When checked, it enables the API Name drop-down list and API Template.
API Name	Applies only to APIs. Enabled through the Invoke an API radio button. Provides a drop-down list of standard or extended APIs available through the Yantra 7x Service Definition Framework. If the API you select is backward compatible, the Requires Backward Compatibility checkbox is enabled.
Requires Backward Compatibility	Applies only to APIs that can be invoked in backward compatibility mode. Enables you to specify if the API should run in backward compatibility mode. When checked, Version must also be supplied.
Version	Applies only to APIs running in backward compatibility mode. Provides a drop-down list of the versions for which an API is backward compatible.

Table 9–9 API Resource Dialog Box

Control	Description
Input Namespace	<p>Namespace corresponding to the text box that passes input to a Save API. Applies only to a detail view, as they may have several text boxes that are bound to different XML namespaces. However, only one of the text boxes can pass input to the API.</p>
Output Namespace	<p>The output of the API is saved in this namespace. Namespace is optional, but if it is not specified, it is defaulted to the root node name of the XML under consideration. Therefore, while referring to the output of the API, even if namespace is not specified here, it can be assumed to be the same as the root node name of the output.</p> <p>A namespace is a tag that can be used to identify a specific XML. The Yantra 7x Presentation Framework enables you to call multiple APIs and store the outputs in different namespaces. In your JSP or in the input to an API, you can refer to values from any namespace that is available at that point.</p>
Ignore Exception	<p>If this API throws an exception then it is not displayed to the user. This option is <i>not</i> available for API resources that are created directly under an Entity resource.</p>
Skip Automatic Execution	<p>When this option is checked, the API is not called automatically when the view is displayed. You can then call this API within the JSP using the <code>callAPI</code> taglib. This option is not available for API resources that are created directly under an Entity resource.</p>
Call In Rollback-Only Mode	<p>Check this box if you want to call this API in rollback-only mode to roll back the changes made in the database.</p> <p>By default, this checkbox is disabled for all Yantra 7x APIs.</p> <p>Note: However, in order to execute this rollback-only operation, a custom action needs to be created. For information about creating custom actions in a screen, refer to the <i>Yantra 7x Customization Guide</i>.</p>

Table 9–9 API Resource Dialog Box

Control	Description
API Type	Specify the type of view from which the API may be invoked. This option is only available for API resources that are created directly under an Entity resource. Such resource types are: <ul style="list-style-type: none"> List - invokes the API from a list view Detail - invokes the API from a detail view
Input	Provide an XML structure that can be used to pass specific input to the API. You can specify dynamic attributes here.
Template	Provide a template XML here. This template XML is passed to the API through the <code>YFSEnvironment</code> class. Although <code>YFSEnvironment</code> class supports passing a complete XML or simply an XML file name, you can only provide a complete XML in this field.

9.3.8 Application Consoles Inner Panel

The Application Consoles Inner Panel Resource Type dialog box enables you to customize inner panels. Inner panels are the UI components that make up detail views.

Table 9–10 Inner Panel Resource Dialog Box

Control	Description
Java Server Page	The JSP file name with full path relative to the <code><YFS_HOME>/webpages</code> directory. Your custom resources typically point to a file within the <code>/extn</code> directory. For example, if you are changing the order detail header inner panel, specify the <code>/extn/om/order/detail/order_header_detail.jsp</code> file. As you populate the <code>/extn</code> directory with custom JSP files, mimic the structure of the directory containing the standard JSPs.
Override Entity ID	For future use.

Table 9–10 Inner Panel Resource Dialog Box

Control	Description
Entity Key Name	For future use.
Template	When the default Detail API is invoked for a view, all the templates specified for each inner panel for the view are merged to form the output template that is passed to the API.

9.3.9 Application Consoles Action

The Application Consoles Action Resource Details dialog box enables you to customize details of an action that can be used in the Yantra 7x Application Consoles.

Table 9–11 Action Resource Dialog Box

Control	Description
Java Server Page	When an action is executed in the user interface, this JSP is called. This can be used to do server-side processing, such as calling multiple APIs.
JavaScript	<p>The JavaScript function specified here is executed through the <code>eval()</code> function. The function body needs to be present in <code>webpages/extn/extn.js</code>. This JS file is included in the <code>container.jsp</code> file and is therefore available in all JSPs.</p> <p>Note: the <code>container.jsp</code> file is one of the Yantra 7x Presentation Framework JSPs. This JSP defines the basic structure of all screens and includes other JSPs, as defined against inner panels, search views, list views and detail views.</p>

Table 9–11 Action Resource Dialog Box

Control	Description
View ID	<p>When a View ID is specified for an Action, the view opens in a modal dialog when user clicks on the action.</p> <p>However, if an API is also configured for the Action, the behavior is as follows: The API is called first, and after the API is called, the current window itself is updated with the view configured here.</p> <p>When a JavaScript is also configured, that's what is called before view or API is called. If the JavaScript returns false, none of the other actions are invoked.</p> <p>If the Action is configured for a list view, the specified View ID opens up in the same browser window, and not in a separate popup.</p>
View Group ID	View group that is navigated to when you choose this action.
Input Name Space	Namespace of the XML that is passed as input to this API if this action calls an API. It is assumed that this namespace exists.
Binding	<p>You can specify an XML Binding here.</p> <p>If the XML attribute returns true or greater than 0 or Y, the action is enabled and the user cannot click on the action. Since it considers all other values disabled.</p>
Selection Key Name	<p>This is the name or the ID attribute of the checkboxes in the inner panel that needs to be checked before this action is clicked up. For example, in an order list screen, only if some orders are selected, you can view the details. Clicking the View Details action without selecting any order results in a client-side error message. This happens because for the View Details action, the resource has been configured to have a selection key name, which is the ID of the checkboxes in the <code>order_list_concise.jsp</code> file.</p> <p>If this field is not specified, it means that the action is not dependent upon any checkbox being checked.</p>

Table 9–11 Action Resource Dialog Box

Control	Description
Input Key Name	<p>If the action being configured requires the user to select records from a list using check boxes within the list, specify the name or ID of the check box object within the JSP here to ensure that the key the check box is associated to is passed as input to the target of this view.</p> <p>For example, the order detail screen contains a service requests inner panel that displays all of the service requests for that order. The cancel action defined on the inner panel requires the user to select one or more service requests (using the check box that appears in the first column of the list). The name of this check box inside the JSP is <code>chkEntityKeyPS</code>. Therefore, to ensure the selected keys are appropriately passed to the API that is called for the cancel action, the Input Key Name of the cancel action is set to <code>chkEntityKeyPS</code>.</p>
Popup	N/A
Close Window On Complete	When this is specified, the current window closes after the current action is performed. For example, use this control for the Save action on a popup window that should close automatically after Save is performed.

9.3.10 Application Consoles Link

The Application Consoles Link Resource Details dialog box enables you to specify the details of a hyperlink used within an inner panel.

Links are not permission controlled. This means that any Links you modify does not show up on the Permission tree of the User Group Configurator.

Table 9–12 Link Resource Dialog Box

Control	Description
View ID	The View ID to open when the link is clicked.
View Group ID	View group that is navigated to when you choose this link.

9.3.11 Application Consoles Icon

The Application Consoles Icon Resource Details dialog box enables you to specify the details of an icon that appears on the upper left section of the inner panel title. These icons are referred to as *view* icons.

Table 9–13 Icon Resource Dialog Box

Control	Description
JavaScript	<p>The JavaScript function specified here is executed through the <code>eval()</code> function. The function body needs to be present in the <code>webpages/extn/extn.js</code> file. This JS file is included in the <code>container.jsp</code> file and is therefore available in all JSPs.</p> <p>Note: The <code>container.jsp</code> file is one of the Yantra 7x Presentation Framework JSP files. This JSP file defines the basic structure of all screens and includes other JSPs, as defined against inner panels, search views, list views and detail views.</p>
View ID	The View ID to open when this icon is clicked.
View Group ID	View group that is navigated to when you choose this icon.
Enabled Binding	<p>You can specify an XML Binding here.</p> <p>If the XML attribute returns true or greater than 0 or Y, the action is enabled and the user cannot click on the action. Since it considers all other values disabled.</p>
Display Binding	<p>You can specify an XML Binding here.</p> <p>If the XML attribute returns true or greater than 0 or Y, the action is enabled and the user cannot click on the action. Since it considers all other values disabled.</p>
Image	Relative path of the image file. For a Yantra 7x Application Consoles menu item, specify the <code>com/yantra/ycp/ui/icons/configmenu.gif</code> file.
Tooltip	<p>You can specify an XML Binding here.</p> <p>When the screen is displayed, the XML pointed to by the binding is evaluated. If the XML attribute returns a numeric value other than zero (0), this number appears in parenthesis next to the resource description in the tooltip of this icon.</p>

Table 9–13 *Icon Resource Dialog Box*

Control	Description
Alternate Image Binding	You can specify an XML Binding here. This field will be used to display a different icon on the console inner panel based on the value of this binding. If the XML attribute returns a value of "Y", "true", or a number greater than 0, then the alternative icon will be used.
Alternate Image	Relative path of the alternate image file.

9.4 Defining Themes

You can define the labels that identify themes used in Yantra 7x. All themes used by Yantra 7x are defined centrally through CSS and XML files. The Themes Configurator contains the standard Yantra 7x themes as well as the ones that you define.

You can use the Themes branch for:

- [Creating a Theme](#)
- [Modifying a Theme](#)
- [Deleting a Theme](#)

9.4.1 Creating a Theme

To create a theme:

1. From the tree in the application rules side panel, choose Presentation > Themes. The Themes window appears in the work area.
2. Choose . The Theme Details pop-up window is displayed.

The image shows a 'Theme Details' dialog box with a title bar and a close button. Inside the dialog, there are three text input fields labeled 'Theme', 'Short Description', and 'Long Description'.

3. In Theme, enter the name of the XML or CSS files, for example, if the theme file is called malachite.xml, enter 'malachite'. Refer to the *Yantra 7x Localization Guide* for naming conventions regarding localization.
4. In Short Description, enter a brief description of the theme.
5. In Long Description, enter a more detailed description of the theme.
6. Choose .

9.4.2 Modifying a Theme

To modify a theme:

1. From the tree in the application rules side panel, choose Presentation > Themes. The Themes window appears in the work area.
2. Select the applicable theme and choose . The Theme Details pop-up window is displayed.
3. In Short Description, enter a brief description of the theme.
4. In Long Description, enter a more detailed description of the theme.
5. Choose .

9.4.3 Deleting a Theme

To delete a theme:

1. From the tree in the application rules side panel, choose Presentation > Themes. The Themes window appears in the work area.
2. Select the applicable theme and choose .

9.5 Defining Custom Common Code Types

Common codes are values that enable a user to choose from a list of options rather than having to enter the data manually. They can be made available to the user from drop-down lists in extended Yantra 7x console screens. For example, if an item is backordered, the user can choose a reason, such as No Stock, from a list. The list of reasons is a list of common codes.

By default, the fields on the standard interface have lists of common codes to which you can add your own common codes.

When adding fields to the user interface, you can provide drop-down lists of custom common codes by adding combo boxes. These common code values can be retrieved within an extended screen using the `getCommonCodeList()` API.

You can use the Custom Common Code Types branch for:

- [Creating a Custom Common Code Type](#)
- [Modifying a Custom Common Code Type](#)
- [Deleting a Custom Common Code Type](#)

9.5.1 Creating a Custom Common Code Type

To create a custom common code type:

1. From the tree in the application rules side panel, choose Presentation > Custom Common Code Types. The Custom Common Code Types window appears in the work area.
2. Choose . The Code Type Details pop-up window is displayed.

The screenshot shows a dialog box titled "Code Type Details". It contains four text input fields arranged vertically:

- Code Type
- Short Description
- Long Description
- Maximum Code Value Length

- In Code Type, enter the name of the custom common code type.

Note: Value of the new common code type Maximum length is 15 characters. This value is automatically appended with `.ex` extension so that it will not conflict with Yantra 7x-defined common codes. This value must be passed to the `getCommonCodeList()` API.

- In Short Description, enter a brief description of the theme.
- In Long Description, enter a more detailed description of the theme.
- In Maximum Code Value Length, enter the maximum length of a code value for this custom common code type. When users create code values for this common code type, they can enter values that have a maximum length of the value specified here.
- Choose .

9.5.2 Modifying a Custom Common Code Type

To modify a custom common code type:

1. From the tree in the application rules side panel, choose Presentation > Custom Common Code Types. The Custom Common Code Types window appears in the work area.
2. Select the applicable custom common code type and choose . The Code Type Details pop-up window is displayed.
3. In Short Description, enter a brief description of the code type.
4. In Long Description, enter a more detailed description of the code type.
5. Choose .

9.5.3 Deleting a Custom Common Code Type

To delete a custom common code type:

1. From the tree in the application rules side panel, choose Presentation > Custom Common Code Types. The Custom Common Code Types window appears in the work area.
2. Select the applicable common code type and choose .

9.6 Defining Custom Common Code Values

You can define values for a custom common code type.

You can use the Custom Common Codes branch for:

- [Adding Values to a Custom Common Code](#)
- [Modifying a Custom Common Code's Values](#)
- [Deleting a Custom Common Code's Values](#)

9.6.1 Adding Values to a Custom Common Code

To add values to a custom common code:

1. From the tree in the application rules side panel, choose Presentation > Custom Common Codes. The Custom Common Code Types window appears in the work area.
2. Select the applicable custom common code and choose . The Common Code Values pop-up window is displayed.

3. Choose . The *(Common Code) Details* pop-up window is displayed.



The screenshot shows a dialog box titled "Common Code 1 Details". It contains three text input fields stacked vertically, labeled "Common Code 1", "Short Description", and "Long Description".

4. In the Common Code Field, enter the code value.
5. In Short Description, enter a brief description of the code value.
6. In Long Description, enter a more detailed description of the code value.
7. Choose .

9.6.2 Modifying a Custom Common Code's Values

To modify a custom common code value:

1. From the tree in the application rules side panel, choose Presentation > Custom Common Codes. The Custom Common Code Types window appears in the work area.
2. Select the applicable custom common code and choose . The Common Codes pop-up window is displayed.
3. Locate the applicable code value and choose . The common code's Details pop-up window appears.
4. In Short Description, enter a brief description of the code value.
5. In Long Description, enter a more detailed description of the code value.
6. Choose .

9.6.3 Deleting a Custom Common Code's Values

To delete a custom common code value:

1. From the tree in the application rules side panel, choose Presentation > Custom Common Codes. The Custom Common Code Types window appears in the work area.
2. Select the applicable custom common code and choose . The Common Codes pop-up window is displayed.
3. Locate the applicable code value and choose .

9.7 Defining Custom Error Codes

You can define custom error codes and the descriptions to be used along with the default error codes provided by Yantra 7x. For example, you can use these custom error codes when the user exit to validate passwords returns validation failure reasons. Moreover, the custom error codes are localizable through existing mechanisms.

All the error codes must start with `ExtN`, if not an exception is thrown when you save the custom error code.

To search for a custom error code

1. From the tree in the application rules side panel, choose Presentation > Custom Error Codes. The Custom Error Code Search window appears in the work area.
2. Enter the applicable error code and description for which you want to search.
3. Choose  and a list of custom error codes displays.

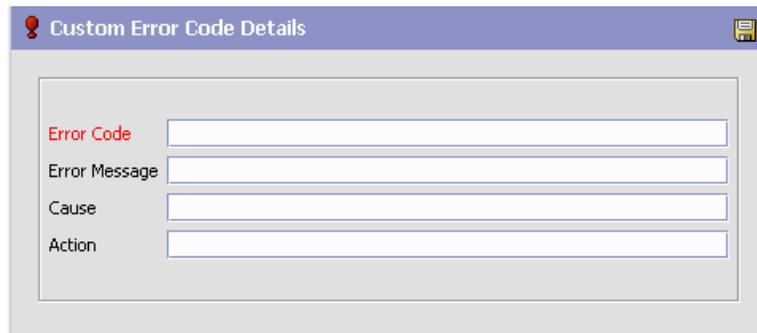
You can use the Custom Error Codes list for:

- [Adding a Custom Error Code](#)
- [Modifying a Custom Error Code](#)
- [Deleting a Custom Error Code](#)

9.7.1 Adding a Custom Error Code

To add values to a custom error code:

1. From the tree in the application rules side panel, choose Presentation > Custom Error Codes. The Custom Error Codes List window appears in the work area.
2. Choose . The Custom Error Code Details pop-up window displays.



3. In the Error Code Field, enter the code value.
4. In Error Message, enter the message to be displayed for this code value.
5. In Cause, enter the cause for the error to be raised.
6. In Action, enter the action to be taken when the error is raised.
7. Choose .

9.7.2 Modifying a Custom Error Code

To modify a custom error code value:

1. From the tree in the application rules side panel, choose Presentation > Custom Error Codes. The Custom Error Code search window appears in the work area.
2. Choose  and a list of custom error codes displays.
3. Select the applicable custom error code from the list and choose . The Error Codes pop-up window displays.
4. In Error Message, modify the message displayed for this code value.

5. In Cause, modify the cause for the error to be raised.
6. In Action, modify the action to be taken when the error is raised.
7. Choose .

9.7.3 Deleting a Custom Error Code

To delete a custom error code value:

1. From the tree in the application rules side panel, choose Presentation > Custom Error Codes. The Custom Error Code search window appears in the work area.
2. Choose  and a list of custom error codes are displayed.
3. Select the code value you want to delete and choose .

10

Configuring Business Communication Components

You can configure business communication components to define the codes and documents used to communicate between Yantra 7x and external systems as well as different business organizations within your business model.

You can use the Communication branch for:

- [Defining Protocol Codes](#)
- [Defining Document Format Codes](#)
- [Defining Business Document Codes](#)

10.1 Defining Protocol Codes

You can use **Protocol Code Setup** to set up codes to identify the different protocols organizations used to communicate with each other.

The following are examples of different protocols:

- FTP
- E-Mail
- Fax

You can use the Protocols branch for:

- [Creating a Protocol Code](#)
- [Modifying a Protocol Code](#)
- [Deleting a Protocol Code](#)

10.1.1 Creating a Protocol Code

To create a protocol code:

1. From the tree in the application rules side panel, choose Communication > Protocols. The Protocol List window appears in the work area.
2. Choose . The Protocol Details pop-up window appears.



3. In Protocol Name, enter the name of the protocol.
4. In Description, enter a brief description of the protocol.
5. Choose .

10.1.2 Modifying a Protocol Code

To modify a protocol code:

1. From the tree in the application rules side panel, choose Communication > Protocols. The Protocol List window appears in the work area.
2. Select the applicable protocol and choose . The Protocol Details pop-up window appears.
3. In Description, enter a brief description of the protocol.
4. Choose .

10.1.3 Deleting a Protocol Code

To delete a protocol code:

1. From the tree in the application rules side panel, choose Communication > Protocols. The Protocol List window appears in the work area.
2. Select the applicable protocol and choose .

10.2 Defining Document Format Codes

You can use **Document Format Code Setup** to set up codes to identify the different document formats organizations use to communicate with each other.

The following are examples of different document formats:

- EDI
- XML
- Flat-file

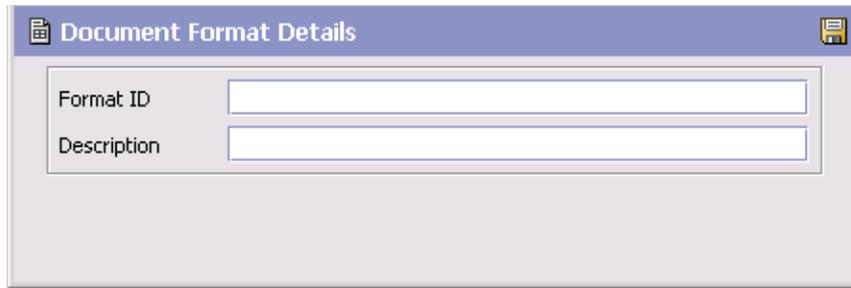
You can use the Document Formats branch for:

- [Creating a Document Format Code](#)
- [Modifying a Document Format Code](#)
- [Deleting a Document Format Code](#)

10.2.1 Creating a Document Format Code

To create a document format code:

1. From the tree in the application rules side panel, choose Communication > Document Formats. The Document Format List window appears in the work area.
2. Choose . The Document Format Details pop-up window appears.



The screenshot shows a dialog box titled "Document Format Details". It contains two text input fields. The first field is labeled "Format ID" and the second is labeled "Description". Both fields are currently empty.

3. In Format ID, enter the name of the document format.
4. In Description, enter a brief description of the document format.
5. Choose .

10.2.2 Modifying a Document Format Code

To modify a document format code:

1. From the tree in the application rules side panel, choose Communication > Document Formats. The Document Format List window appears in the work area.
2. Select the applicable document format and choose . The Document Format Details pop-up window appears.
3. In Description, enter a brief description of the document format.
4. Choose .

10.2.3 Deleting a Document Format Code

To delete a document format code:

1. From the tree in the application rules side panel, choose Communication > Document Formats. The Document Format List window appears in the work area.
2. Select the applicable document format and choose .

10.3 Defining Business Document Codes

You can use **Business Document Code Setup** to set up codes to identify the different documents organizations use to communicate with each other.

The following are examples of different document formats:

- Purchase Order (PO)
- Advanced Shipment Notice (ASN)
- Invoice

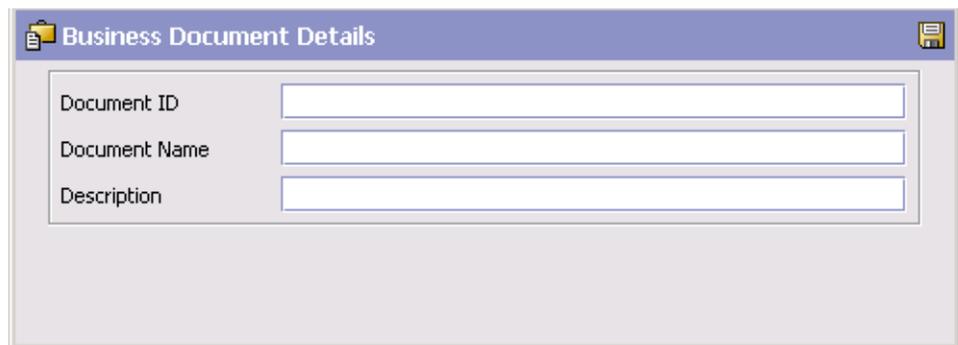
You can use the Business Documents branch for:

- [Creating a Business Document Code](#)
- [Modifying a Business Document Code](#)
- [Deleting a Business Document Code](#)
- [Classifying an Existing Business Document](#)

10.3.1 Creating a Business Document Code

To create a business document code:

1. From the tree in the application rules side panel, choose Communication > Business Document. The Business Document List window appears in the work area.
2. Choose  from the top portion of the screen. The Business Document Details pop-up window appears.



The screenshot shows a window titled "Business Document Details" with a light blue header and a light gray body. The window contains three input fields stacked vertically, each with a label to its left: "Document ID", "Document Name", and "Description". The input fields are empty and have a light blue border. There is a small icon in the top right corner of the window.

3. In Document ID, enter the ID of the business document.
4. In Document Name, enter the name of the business document.
5. In Description, enter a brief description of the document format.
6. Choose .

10.3.2 Modifying a Business Document Code

To modify a business document code:

1. From the tree in the application rules side panel, choose Communication > Business Document. The Business Document List window appears in the work area.
2. Select the applicable business document and choose . The Business Document Details pop-up window appears.
3. In Document Name, enter the name of the business document.
4. In Description, enter a brief description of the document format.
5. Choose .

10.3.3 Deleting a Business Document Code

To delete a business document code:

1. From the tree in the application rules side panel, choose Communication > Business Document. The Business Document List window appears in the work area.
2. Select the applicable document format and choose .

10.3.4 Classifying an Existing Business Document

You can classify any existing business documents to be identified as a Buyer, Seller, or Carrier Document. For example, an advanced shipment notice is a document sent to carriers to alert them that an order has been made and a shipment of a certain set of items is necessary. You would identify this document as a carrier document.

To classify an existing business document as a Buyer, Seller, or Carrier document:

1. From the tree in the application rules side panel, choose Communication > Business Document. The Business Document List window appears in the work area.
2. Choose either the Seller Documents tab, Buyer Documents tab, or Carrier Documents tab dependant on which role you want to associate the business document with.
3. Choose . The Role Document Details pop-up window appears.



The screenshot shows a window titled "Role Document Details". It contains the following fields:

- Role:** A dropdown menu with "Seller" selected.
- Document ID:** A dropdown menu with a document icon on the right.
- Document Name:** A text input field.
- Description:** A text input field.

4. From Document ID, select the business document you want to associate with the role.
5. In Document Name, enter the name of the business document.
6. In Description, enter a brief description of the document format.
7. Choose .

Configuring Nomenclature Components

The Nomenclature Runtime components provide a mapping tool that allows you to configure unique terms you use to match unique terms your trading partners use.

When trading partners or external systems exchange information with Yantra 7x, they may identify entity values differently than Yantra 7x and each other. For example, a carrier code may be called `UPS` in Yantra 7x and `UnitedParcelService` by a trading partner or external system.

The Nomenclature Transformation Engine enables you to define these entities, its values, and the mappings between interacting systems or trading partners. Then, when data is exchanged between Yantra 7x and an external system or trading partner, the Nomenclature Transformation Engine automatically applies the mapping rules specified.

Using the United Parcel Service example above, when data is sent from Yantra 7x to the external system or trading partner, the Nomenclature Transformation Engine transforms the carrier code literal `UPS` to `UnitedParcelService` and back again.

You can use the Nomenclature branch for:

- [Defining Nomenclature Codes](#)
- [Defining Nomenclature Definitions](#)
- [Defining Nomenclature Configuration](#)

11.1 Defining Nomenclature Codes

Nomenclature Codes allows you to create, modify, and delete attribute codes that can be used to define entities in Nomenclature Definition.

To create a nomenclature code:

1. From the menu bar, choose Applications > Platform. The Platform tree is displayed in the side panel.
2. From the Platform tree, choose Nomenclature > Nomenclature Codes. The Nomenclature Codes window appears in the work area.
3. Choose . The Nomenclature Code Details pop-up window is displayed.



The screenshot shows a dialog box titled "Nomenclature Code Details". It has a title bar with the text "Nomenclature Code Details" and a small icon on the right. The main area contains three text input fields, each with a label to its left: "Nomenclature Code", "Short Description", and "Long Description".

4. In Nomenclature Code, enter the name of the attribute code.
5. In Short Description, enter a brief description of the nomenclature code.
6. In Long Description, enter a detailed description of the nomenclature code.
7. Choose .

11.2 Defining Nomenclature Definitions

Nomenclature Definition allows for entities requiring transformation to be created. These entities are identified for each System and Participant (Enterprise, Buyer, Seller, Carrier). Entities definitions are created for a combination of one or more attribute codes. The attribute codes that make an entity are common across all participating systems and participants.

To create a nomenclature definition:

1. From the tree in the application rules side panel, choose Nomenclature > Nomenclature Definition. The Nomenclature Definition Search window appears in the work area.
2. Choose . The Nomenclature definition details pop-up window is displayed. Enter information, using the [Table 11–1, "Nomenclature Definition Menu"](#) for help.
3. Choose .

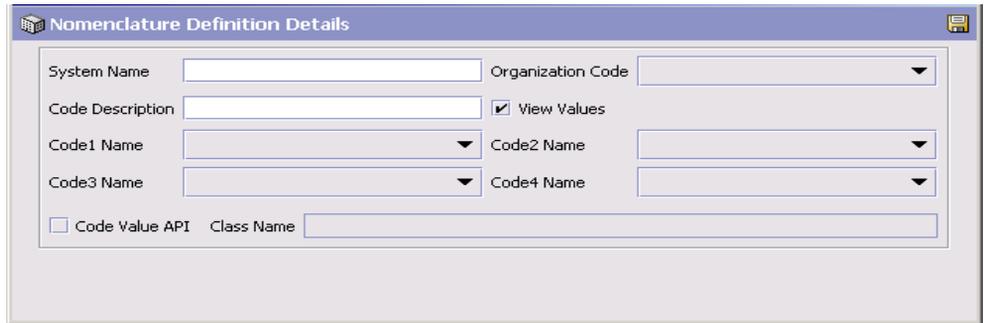


Table 11–1 Nomenclature Definition Menu

Property	Description
System Name	Enter a unique System Name for each system you are defining this entity for. This is a required parameter. The System Name YANTRA is reserved.
Organization Code	Select a participant for Organization Code. Select the DEFAULT organization from the drop down list if the definition applies across participants. This is a required parameter. The Organization code uniquely identifies the Trading partner or the external system.
Code Description	Enter a unique description for the entity being created.
View Values	Defaults to view values. Check this if you do not want to see a list of valid values for a System and Participant. Valid values for the entity are not displayed when mapping if this is not checked.

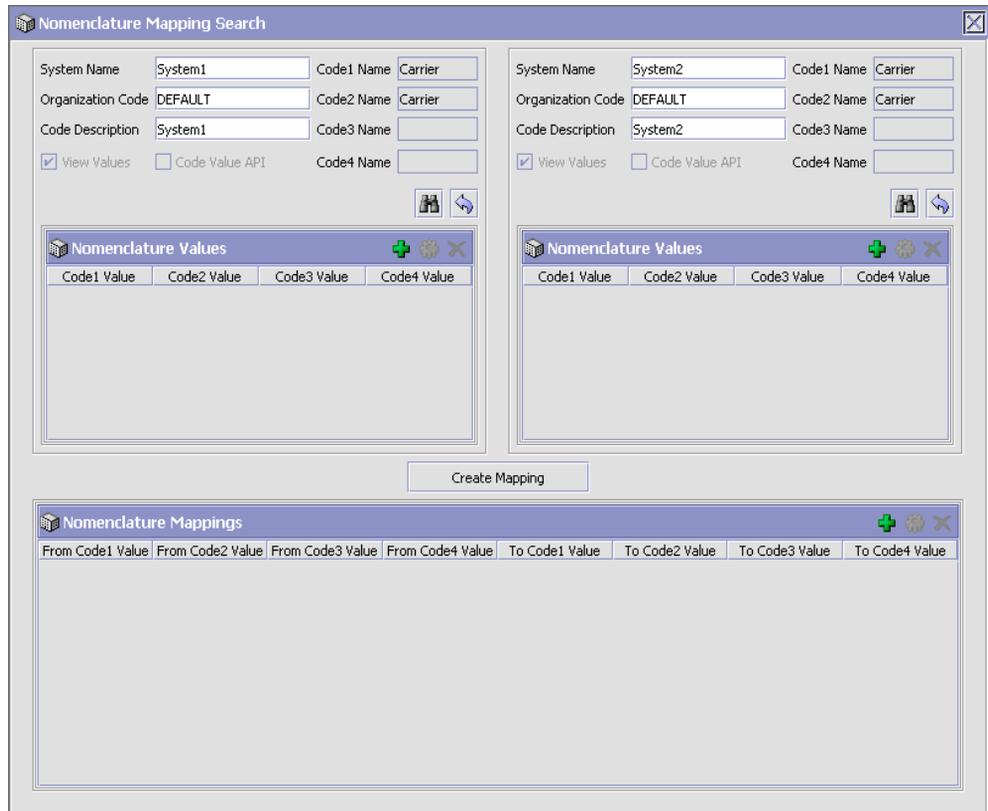
Table 11–1 Nomenclature Definition Menu

Property	Description
Code Name 1-4	Select up to four attribute codes that can be grouped together to define the entity. A minimum of one attribute must be selected. Additional attributes may be added to the list through Nomenclature Code Maintenance. The list is displayed from the YFS_COMMON_CODE table for CODE_TYPE='XREF_CODE'.
Code Value API	Check on the code value API box and enter a class name, which retrieves the list of values for the code.
Class Name	Enter the class name used to get values from the external system. The system has pre-defined classes for certain attribute codes, which retrieve data for the System Name YANTRA. (Yantra 7x defined code values) Using a class to get a list of valid value for an attribute code is not mandatory. The code values can be created and mapped directly when mapping entities.

11.2.1 Creating Mappings between Nomenclature Definitions

To create mapping between nomenclature definitions:

1. From the tree in the application rules side panel, choose Nomenclature > Nomenclature Definition. The Nomenclature Definition Search window appears in the work area.
2. Select the two Entities that you want to map and choose the Mapping button. The Nomenclature Mapping panel is displayed.



The Nomenclature Mapping panel has the list of valid values for the entities selected. The from/to Entity to be mapped can be changed by changing the System Name, Organization Code, or the Entity description.

Note: The list is not displayed if the View Values option is not checked when creating the Entity Definition.

3. Choose . The new entity defined is added to the list of entities in the search results.
4. Enter values for each attribute and choose .

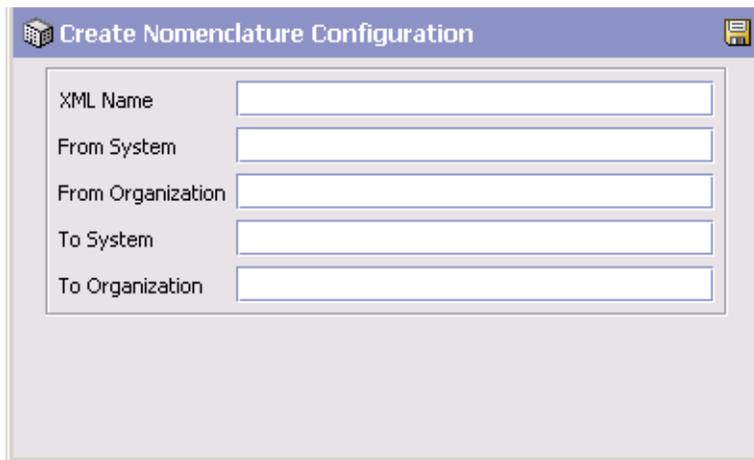
5. To create a mapping between the two entities, select a row from the list of values displayed in the from/to system and choose the Create Mapping button.

11.3 Defining Nomenclature Configuration

Nomenclature Configuration allows you to define the rules to be applied and the entities that need to be transformed when data is exchanged between Yantra 7x and external systems or trading partners. The configuration is captured for each document exchanged.

To create a unique configuration between two systems:

1. From the tree in the application rules side panel, choose Nomenclature > Nomenclature Configuration. The Nomenclature Configuration Search window appears in the work area.
2. Choose . The Create Nomenclature Configuration pop-up window is displayed.
3. Enter information in the applicable fields. Refer to [Table 11–2, "Nomenclature Configuration"](#) for field level descriptions.

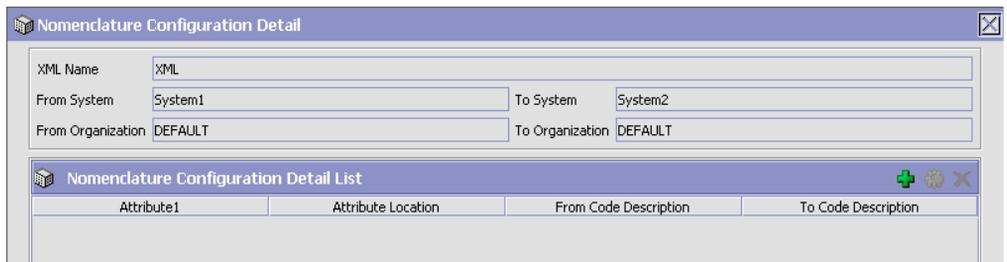


The screenshot shows a dialog box titled "Create Nomenclature Configuration". The dialog has a title bar with a cube icon on the left and a save icon on the right. The main area contains five text input fields, each with a label to its left: "XML Name", "From System", "From Organization", "To System", and "To Organization".

Table 11–2 Nomenclature Configuration

Property	Description
XML Name	Enter a unique name for the document being exchanged between two systems or participants.
From System	Enter the system name from which document is originating.
To System	Enter the system name to which the document is delivered to
From Organization	Enter the organization from which the document is originating.
To Organization	Enter the organization to which the document is delivered to

4. Choose . The Configuration Details window is displayed.



5. Choose . The Configuration Detail pop-up window appears capturing all the fields in the document that need to be transformed.
6. Enter information in the applicable fields. Refer to [Table 11–3, "Nomenclature Configuration Details"](#) for field level descriptions.

Table 11–3 Nomenclature Configuration Details

Property	Description
XML Name	The unique name by which the document exchanged is identified.
Attribute Location	<p>Enter the full XML path of the element under which the attributes that need transformation are present.</p> <p>If this is a repeating node in the XML document, the transformation is applied to all the nodes.</p> <p>For example, if the <code>publishShipAdvice</code> output XML is being sent to a warehouse management system for fulfillment and the carrier code needs to be transformed, the attribute location should be entered as <code>ShipmentAdvices/ShipmentAdvice</code>.</p>

Table 11–3 Nomenclature Configuration Details

Property	Description
Attribute 1-4	<p>Enter the XML attribute name(s) that need to be transformed when the document is being published.</p> <p>For example, if the <code>publishShipAdvice</code> output XML is being sent to a warehouse management system for fulfillment and the carrier code needs to be transformed, the attribute should be entered as <code>Scac</code> in the document location <code>ShipmentAdvices > ShipmentAdvice</code>.</p>
New Attribute 1-4	<p>For each attribute being transformed, either the existing value can be replaced in the document or a new attribute can be inserted in the XML in the same level. Enter the new XML attribute name to be inserted when the transformation occurs.</p> <p>For example, the <code>publishShipAdvice</code> output XML is being sent to a warehouse management system for fulfillment and the carrier code needs to be transformed in the location <code>ShipmentAdvices > ShipmentAdvice</code> in the document for the SCAC attribute. This is specified as <code>WhseCarrier</code> and a new attribute is inserted in the location <code>ShipmentAdvices > ShipmentAdvice</code> called <code>WhseCarrier</code>. This XML attribute carries the new transformed value and the old attribute and value are left intact.</p>
Mapping Type	<p>Select Nomenclature if the values are retrieved from the mapping specified in the Nomenclature transformation Engine.</p> <p>Select Constant if the value is always transformed to a Constant value.</p>
Abort Processing on Error	<p>If this is selected and the Mapping Type is set to nomenclature, if mapping is not found a 'No Default' is set and processing stops.</p>
Nomenclature	<p>Specifies the entity definition to use for the system participant to determine the transformed values. Select the from/to code descriptions to use.</p>
Use Default Values	<p>If this is selected, enter the default values to apply when mapping is not found.</p>
Constant 1-4	<p>If you selected Constant, enter the constant value for XML attribute transformation.</p>

Configuring Alert Queues

Yantra 7x Queue Management is used to define rules and methods pertaining to user alert notifications.

Alerts

An **alert** is a message directed to a user or an alert queue about a transaction that needs manual intervention. An alert can come in different formats including e-mail, faxes, and so on.

Alerts are sent to different queues depending on the notification definitions you have configured.

Alert Queues

Alert queues are set up to distribute alerts to users. You determine which users receive different alert types by assigning them to queues. You can also set up alert priorities and actions raised when certain conditions are met for the alert.

You can use the Queue Management branch for:

- [Creating an Alert Queue](#)
- [Modifying an Alert Queue](#)
- [Deleting an Alert Queue](#)

12.1 Creating an Alert Queue

To create an alert queue:

1. From the tree in the application rules side panel, choose Queue Management. The Queue Search window appears in the work area.

2. Choose . The Queue Details window is displayed.
3. Enter information in the applicable fields. Refer to [Table 12–1](#) for field value descriptions.

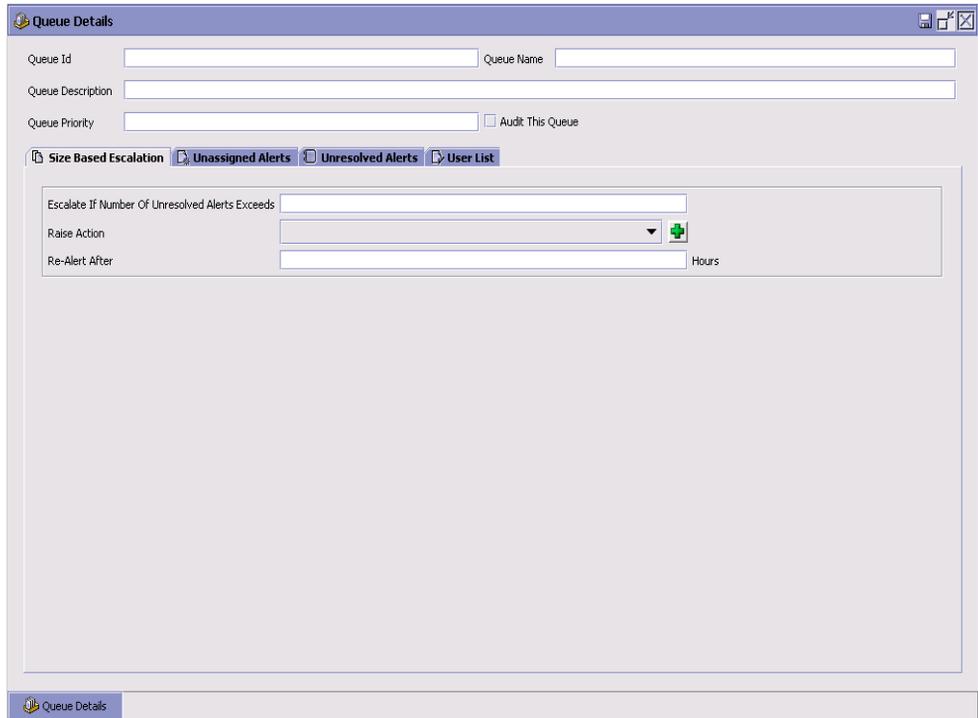


Table 12–1 *Queue Details Window*

Field	Description
Queue ID	Enter a unique identifier for the queue.
Queue Name	Enter the name of the queue.
Queue Description	Enter a brief description of the queue.

Table 12–1 Queue Details Window

Field	Description
Queue Priority	Enter a numerical priority for the queue with 1 being the highest and 0 being no priority. This is used to identify a queue's importance in the business environment.
Audit Queue	Select Audit Queue if you want to audit the alerts coming into the queue and how they are resolved.

You can use the Queue Details window for:

- [Defining How Unresolved Alerts are Handled Based on Size](#)
- [Defining How Unassigned Alerts are Handled Based on Time](#)
- [Defining How Unresolved Alerts are Handled Based on Time](#)
- [Viewing an Alert Queue's User List](#)
- [Modifying an Alert Queue](#)

12.1.1 Defining How Unresolved Alerts are Handled Based on Size

You can configure unresolved alerts to be escalated to a different queue after a specified number of unresolved alerts has been exceeded.

To set up a size-based escalation:

1. In the Queue Details window, choose the Size Based Escalation tab.
2. Enter information in the applicable fields. Refer to [Table 12–2](#) for field value descriptions.

Table 12–2 Size Based Escalation Tab

Field	Description
Escalate If Number of Unresolved Alerts Exceeds	Enter the maximum number of unresolved alerts that can be logged in this queue before an action is raised.
Raise Action	Select an action to be taken when the number of unresolved alerts in this queue is greater or equal to the specified maximum.
Re-Alert After (in hours)	Enter the number of elapsed hours before a re-alert is generated.

12.1.2 Defining How Unassigned Alerts are Handled Based on Time

You can configure alerts that have not been assigned to a user to be escalated to another queue, raise an action after a given amount of time passes, or both.

To set up a time-based escalation for unassigned alerts:

1. In the Queue Details window, choose the Unassigned Alerts tab.
2. Enter information in the applicable fields. Refer to [Table 12–3](#) for field value descriptions.

Table 12–3 Unassigned Alerts Tab

Field	Description
Escalate Alert After (in hours)	Enter the maximum number of hours after which an unassigned alert is moved to another queue.
Move Alert To Different Queue	Select the name of the queue where unassigned alerts should be moved, typically a higher priority queue.

Table 12–3 Unassigned Alerts Tab

Field	Description
Raise Action	Select the action to be raised, if applicable.
Re-Alert After (in hours)	Enter the number of elapsed hours before a re-alert is generated.

12.1.3 Defining How Unresolved Alerts are Handled Based on Time

You can configure unresolved alerts to be escalated to another queue, raise an action after a given amount of time passes, or both.

To set up a time-based escalation for unresolved alerts:

1. In the Queue Details window, choose the Unresolved Alerts tab.
2. Enter information in the applicable fields. Refer to [Table 12–4](#) for field value descriptions.

Table 12–4 Unresolved Alerts Tab

Field	Description
Escalate Alert After (in hours)	Enter the maximum number of hours an alert can remain unresolved in this queue.
Move Alert To Different Queue	Select the name of the queue where unresolved alerts should be moved, typically a higher priority queue.
Raise Action	Select the action to be raised, if applicable.
Re-Alert After (in hours)	Enter the number of elapsed hours before a re-alert is generated.

12.1.4 Viewing an Alert Queue's User List

You can view the users that are subscribed to an alert queue and modify their subscription details. For more information about users, see [Section 5.2, "Defining Users"](#) on page 230.

To view and modify users subscribed to a queue:

1. In the Queue Details window, choose the User List tab. The User Detail list is displayed.
2. Select the applicable user and choose . The User Detail window is displayed. For more information about how to modify a user, see [Section 5.2.1, "Creating and Modifying a User"](#) on page 230.)
3. To remove a user from the queue choose .



User ID	User Name	Organization
Barb	Barbara Ann	DEFAULT
yantra	Administrator	DEFAULT

12.2 Modifying an Alert Queue

Once an alert queue has been defined, it can be modified.

To modify an alert queue:

1. From the tree in the application rules side panel, choose Queue Management. The Queue Search window appears in the work area.
2. Enter applicable search criteria and choose . The Queue list is displayed.
3. Select the applicable queue and choose . The Queue Details window is displayed.
4. Refer to the topics under [Section 12.1, "Creating an Alert Queue"](#) for assistance.

12.3 Deleting an Alert Queue

To delete an alert queue:

1. From the tree in the application rules side panel, choose Queue Management. The Queue Search window appears in the work area.
2. Enter applicable search criteria and choose . The Queue list is displayed.
3. Select the applicable queue and choose .

Configuring Region Definitions

Region definitions allows you to configure the components that are used by Yantra 7x geography engines. The individual components consisting of regions and region levels can be used to create region schemas that can then be used throughout the Yantra 7x business application models whenever geography is considered (for example, when determining the regions a delivery service delivers to).

You can use the Region Definition branch for:

- [Defining Region Levels](#)
- [Defining Region Match Preferences](#)
- [Defining Region Schemas](#)

13.1 Defining Region Levels

A region level classifies regions into distinct categories. You can define region levels such as Country, State, County, City, and so on, based on the levels at which you want to aggregate your regions, and define the address field a region level corresponds to. Region levels also allow users to create a region hierarchy.

You can use the Region Levels branch for:

- [Creating a Region Level](#)
- [Modifying a Region Level](#)
- [Deleting a Region Level](#)

13.1.1 Creating a Region Level

To create a region level:

1. From the tree in the application rules side panel, choose Region Definition > Region Levels. The Region Levels window appears in the work area.
2. Choose . The Region Level Details window displays.
3. Enter information in the applicable fields. Refer to [Table 13–1](#) for field value descriptions.
4. Choose .

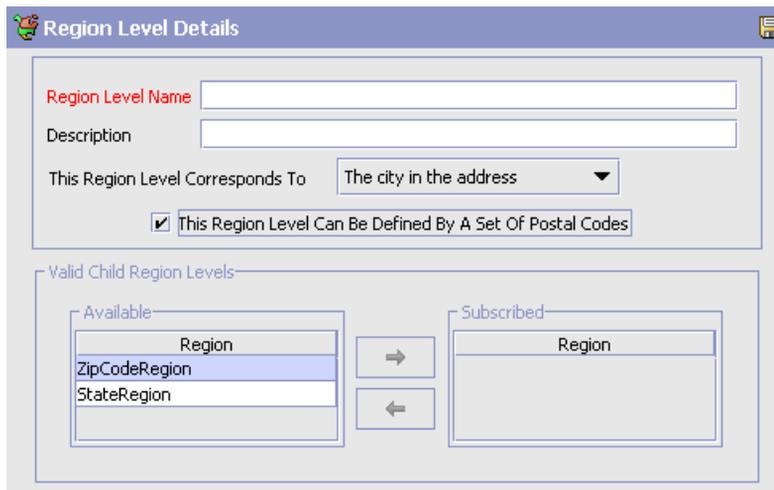


Table 13–1 *Region Level Details Window*

Field	Description
Region Level Name	Enter the name of the region level.
Description	Enter a brief description of the region level.
This Region Level Corresponds To	Select the corresponding address field for this region level from the drop-down list. For more information on region matching, refer to Section 13.2, "Defining Region Match Preferences" on page 356.

Table 13–1 Region Level Details Window

Field	Description
This Region Level Can Be Defined By A Set Of Postal Codes	Select This Region Level Can Be Defined By A Set Of Postal Codes if you want to indicate that regions associated with this region level can define a postal code range. This is used to default the Postal Codes Define This Region field when defining a region for this region level only.
Valid Child Region Levels	If this region has any child region levels that have already been defined, select the applicable region levels from the Available table and choose the right-arrow button.

Note: We recommend creating region levels from the lowest region level to the highest. For example, you create a region level called Town. Then you create a region level called County. Since Town was created earlier you can move it under County at the same time you are creating the County region level.

If you had created County first, you would have had to close it, then create Town, then close Town and reopen County to add Town beneath it.

13.1.2 Modifying a Region Level

To modify a region level:

1. From the tree in the application rules side panel, choose Region Definition > Region Levels. The Region Levels window appears in the work area.
2. Select the applicable region level and choose . The Region Level Details window displays.
3. Enter information in the applicable fields. Refer to [Table 13–1](#) for field value descriptions.
4. Choose .

13.1.3 Deleting a Region Level

To delete a region level:

1. From the tree in the application rules side panel, choose Region Definition > Region Levels. The Region Levels window appears in the work area.
2. Select the applicable region level and choose .

13.2 Defining Region Match Preferences

Region match preferences allow you to specify the level at which addresses should be matched to regions, for each country. For more information on region matching, see *Yantra 7x Product Concepts*.

You can use the Region Match Preferences branch for:

- [Setting a Region Match Preference](#)
- [Deleting a Region Match Preference](#)

13.2.1 Setting a Region Match Preference

To set a region match preference:

1. From the tree in the application rules side panel, choose Region Definition > Region Match Preferences. The Region Match Preferences window appears in the work area.
2. Enter information in the applicable fields. Refer to [Table 13–2](#) for field value descriptions.
3. Choose .

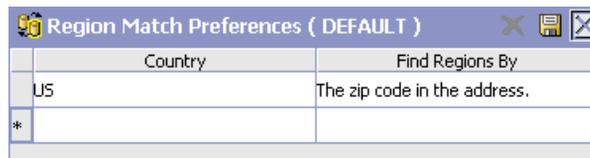


Table 13–2 Region Match Preferences

Field	Description
Country	Select the country code from the drop-down list.
Find Regions By	Select the address field to match regions to for the specified country.

13.2.2 Deleting a Region Match Preference

To delete a region match preference

1. From the tree in the application rules side panel, choose Region Definition > Region Match Preferences. The Region Match Preferences window appears in the work area.
2. Select the applicable region match preference and choose .

13.3 Defining Region Schemas

A region schema is the complete hierarchical set of regions that define a given geography. A region is configured as a specific territory. For example, you can create a region for a complete state, city, or town.

You can create a region hierarchy by defining certain regions as a parent region of a smaller region.

You can use the Region Schemas branch for:

- [Creating a Region Schema](#)
- [Modifying a Region Schema](#)
- [Deleting a Region Schema](#)

13.3.1 Creating a Region Schema

To create a region schema:

Important: Region schemas should be created from the top down. For example, if your region schema consists of country, state, and city regions, you need to create the country region first, then states, followed by cities.

1. From the tree in the application rules side panel, choose Region Definition > Region Schemas. The Region Schemas window appears in the work area.
2. Choose . The Region Schema Details window displays.
3. Enter information in the applicable fields. Refer to [Table 13–3](#) for field value descriptions.
4. Choose .

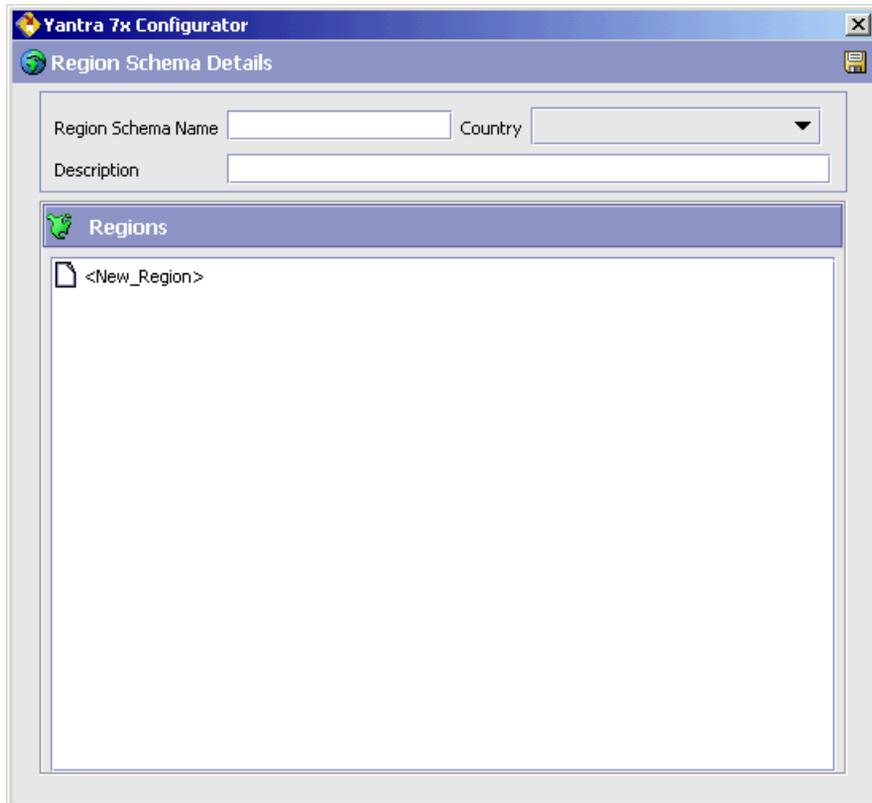


Table 13–3 *Region Schema Details Window*

Field	Description
Region Schema Name	Enter the name of the region schema.
Country	Select the country in which the region schema is located (can be the whole country, a territory in the country, and so forth). This field is optional and is only used for defaulting a country when entering postal code ranges when defining an individual region.
Description	Enter a brief description of the region schema.
Regions	A graphical representation of the region hierarchy.

You can use the Region Schema Details window for:

- [Creating a Region](#)
- [Modifying a Region](#)
- [Deleting a Region](#)

13.3.1.1 Creating a Region

To create a region:

1. In the Region Schema Details window, highlight a region in the region hierarchy under which you want to add a new region and choose . The Region Details window is displayed.
2. Enter information in the applicable fields. Refer to [Table 13–4](#) for field value descriptions.
3. Choose .

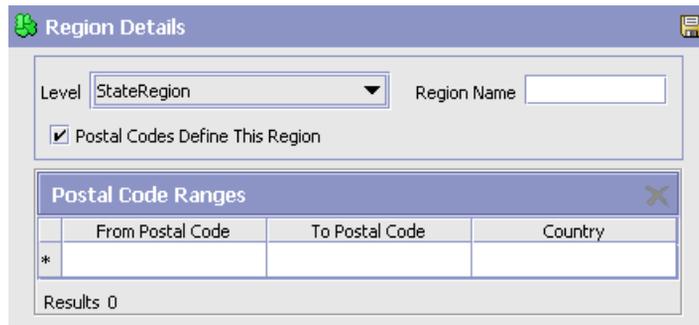


Table 13–4 *Region Details Window*

Field	Description
Level	Select the region's level. For more information about region levels, see Section 13.1, "Defining Region Levels" on page 353.
Region Name	Enter the name of the region. If a region level has been selected that maps to the 'Country' field, this field will provide a drop-down of available country codes to select from.

Table 13–4 Region Details Window

Field	Description
Postal Codes Define This Region	<p>Check this box if you want to define this region by one or more postal codes.</p> <p>Checking this box enables the Postal Code Ranges table.</p> <p>If the region level you have selected for this region does not have the "This Region Level Can Be Defined By A Set Of Postal Codes" option checked, then this checkbox is disabled.</p> <p>For more information of defining region levels, see Section 13.1, "Defining Region Levels" on page 353.</p>
Postal Code Ranges	<p>If you selected Postal Codes Define This Region, enter the postal code range of the region you are configuring and select the country in which the postal codes are defined for.</p>

13.3.1.2 Modifying a Region

To modify a region:

1. In the Region Schema Details window, select the applicable region from the region hierarchy and choose . The Region Details window is displayed.
2. Enter information in the applicable fields. Refer to [Table 13–4](#) for field value descriptions.
3. Choose .

13.3.1.3 Deleting a Region

To delete a region, select the applicable region in the Region Details window and choose .

Note: All child regions are also deleted.

13.3.2 Modifying a Region Schema

To modify a region schema:

1. From the tree in the application rules side panel, choose Region Definition > Region Schemas. The Region Schemas window appears in the work area.
2. Select the applicable region schema and choose . The Region Schema Details window displays.
3. Enter information in the applicable fields. Refer to [Table 13–3](#) for field value descriptions.
4. Choose .

13.3.3 Deleting a Region Schema

To modify a region schema:

1. From the tree in the application rules side panel, choose Region Definition > Region Schemas. The Region Schemas window appears in the work area.
2. Select the applicable region schema and choose .

14

Configuring Devices

A warehouse consists of a number of hand-held and stationary devices. These devices have their unique definitions and sometimes are associated specifically to stations or equipment. Examples of devices include printer, RF scanner and weighing scale.

Each individual group of devices is represented as a device type and sub-type combination. A device and its unique communication requirements are represented when each device is configured.

Use Devices to set-up:

- [Defining a Device Type](#)
- [Defining a Device Sub Type](#)
- [Defining a Device](#)

14.1 Defining a Device Type

All devices are associated with a Device Type in Yantra 7x. An individual unit is defined as a sub type for a device type.

For example, device types include RF scanners, printers, and weighing scale.

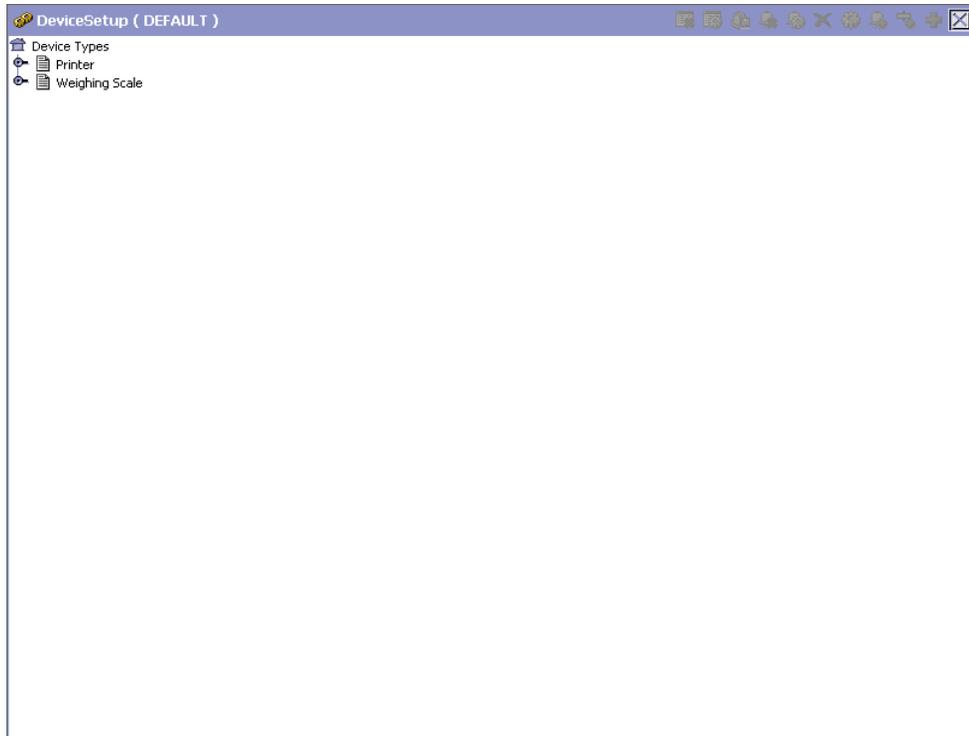
Use Device Type for:

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

14.1.1 Creating a Device Type

To create a device type:

1. From the tree in the application rules side panel, choose Device. The Device Setup window appears.



2. In the Device Setup window, choose . The Device Type pop-up window is displayed.
3. Enter information in the applicable fields. Refer [Table 14–1](#) for field value descriptions.
4. Choose .

The screenshot shows a window titled "Device Type" with a blue header bar. Below the header, there are two text input fields. The first field is labeled "Device Type" and the second is labeled "Description". The window has a standard Windows-style title bar with a close button on the right.

Table 14–1 *Device Type Pop-up Window*

Field	Description
Device Type	Enter a name for the device type. This helps in identifying the type of device. For example, device type may be weighing scale or printer.
Description	Enter a brief description for the device type.

14.1.2 Modifying a Device Type

Once a Device Type has been created, it can be modified.

To modify a device type:

1. From the tree in the application rules side panel, choose Device.
2. The Device Setup window appears with the list of Device Types.
3. Select the Device Type to be modified. Choose .
4. The Device Type pop-up window appears.
5. Enter information in the applicable fields. Refer [Table 14–1](#) for field value descriptions.
6. Choose .

14.1.3 Deleting a Device Type

To delete a device type:

1. From the tree in the application rules side panel, choose Devices.
2. The Device Setup window appears with the list of Device Types.
3. Select the Device Type to be deleted. Choose .

14.2 Defining a Device Sub Type

A Device Sub Type categorizes a device type.

For example, a device type of Printers is further categorized or sub-typed into HP LaserJet 5P, and Zebra 170. Each individual sub-type allows for device configuration and its respective parameters.

Other examples include sub types of hand-held scanner models and equipment mounted models used under a device type of RF Scanners.

Use Device Sub Type for:

- [Creating a Device Sub Type](#)
- [Modifying a Device Sub Type](#)
- [Deleting a Device Sub Type](#)

14.2.1 Creating a Device Sub Type

To create a device sub type:

1. From the tree in the application rules side panel, choose Device. The Device Setup window appears.
2. In the Device Setup window, select Device Type whose Device Sub Type is to be created.
3. Choose . The Device Sub Type pop-up window appears.
4. Enter information in the applicable fields. Refer [Table 14–2](#) for field value descriptions.
5. Choose .

Table 14–2 *Device Sub Type Pop-up Window*

Field	Description
Device Type	Device Type indicates the device type for which the device sub type is being created. This is populated by the system, based on the selection of device type in the Device Setup window.
Device Sub Type	Enter a name for the device sub type.
Description	Enter a brief description for the device sub type.

14.2.2 Modifying a Device Sub Type

Once a Device Sub Type has been created, it can be modified.

To modify a device sub type:

1. From the tree in the application rules side panel, choose Device. The Device Setup window appears.

2. In the Device Setup window, select the Device Type whose Device Sub Type is to be modified. The list of Device Sub Type is now displayed.
3. Select the Device Sub Type to be modified. Choose .
4. The Device Sub Type pop-up window appears.
5. Enter information in the applicable fields. Refer [Table 14–2](#) for field value descriptions.
6. Choose .

14.2.3 Deleting a Device Sub Type

To delete a device sub type:

1. From the tree in the application rules side panel, choose Device. The Device Setup window appears.
2. In the Device Setup window, select the Device Type whose Device Sub Type is to be deleted. The list of Device Sub Type is now displayed.
3. Select the Device Sub Type to be deleted.
4. Choose .

14.3 Defining a Device

A device represents an actual device existing on the network, or directly connected to a station or equipment. All instances of a device type and sub-type combination require to be defined as devices.

For example, a warehouse that has five HP LaserJet 5P printers and four Zebra R140 printers will have all the nine printers configured as devices.

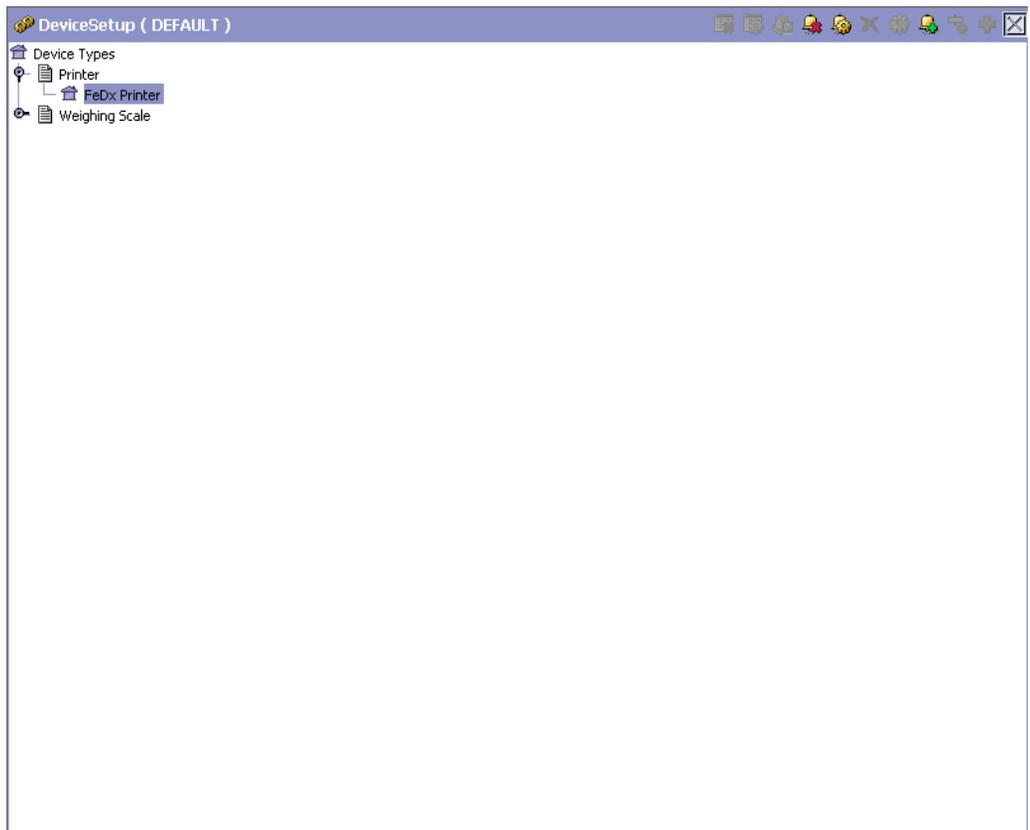
Use Device for:

- [Creating a Device](#)
- [Creating a New Device from a Device](#)
- [Modifying a Device](#)
- [Deleting a Device](#)

14.3.1 Creating a Device

To create a device:

1. From the tree in the application rules side panel, choose Device. The Device Setup window appears.
2. In the Device Setup window, select the relevant Device Type and Device Sub Type whose Device is to be created.



3. Choose . The Device pop-up window appears.
4. Enter information in the applicable fields. Refer [Table 14–2](#) for field value descriptions.
5. Choose .

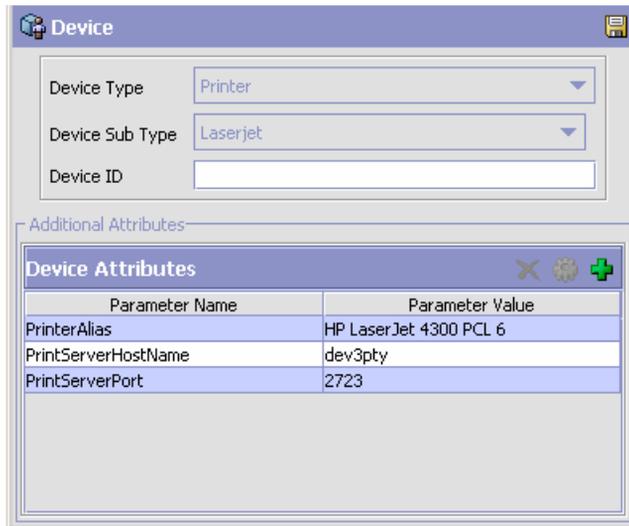


Table 14–3 Device Pop-up Window

Field	Description
Device Type	Device Type indicates the device type for which the device is being created. This is populated by the system, based on the selection of device type in the Device Setup window.
Device Sub Type	Device Sub Type indicates the device sub type for which the device is being created. This is populated by the system, based on the selection of device sub type in the Device Setup window.
Device ID	Enter the name for the device. This identifies the device throughout the system.
Device Attributes	This indicates the additional attributes of the device. For more information about setting up a device attribute, refer to Section 14.3.1.1, "Setting Up a Device Attribute" on page 372.

Yantra 7x supplies a list of standard device type, sub type and individual devices that is supported. The definition of a new device type, sub type and resultant device requires the creation of the appropriate attributes that define the communication with the device.

The list of attributes that control communication to a printer are:

- **DropDirectory** - The directory where the print files are 'dropped' by Yantra 7x Application Server. The Software Print Server keeps polling this directory to pick up print requests. When mentioning the directory structure you can use the full path name or replace the path name with a variable. For more information on including this variable see *Yantra 7x Customization Guide*.

Note: The DropDirectory attribute will appear in the Yantra 7x Configurator only if the property `yfs.software.tcpip.sockets` defined in `yfs.properties` is set to 'N'.

- **PrinterAlias** - The printer alias as configured in the Software printer setup.

Note: While setting up a Printer device in Yantra 7x, please ensure that the Printer Alias is exactly the same as specified in the Software printer set-up.

In instances where a network printer is used, please ensure that the Printer Alias on Yantra 7x does NOT contain the prefix "\\". However, Software may require the printer to be defined by prefixing "\\".

- **PrinterServerHostName** - The host name for the Software Print Server. While IP Address may be sufficient, the use of host name is recommended for ease of maintenance.
- **PrintServerPort** - The port on which Software Print Server listens for print requests. By default, the print server port for Software Print Server is 2723.

Note: The PrinterServerHostName and PrintServerPort attributes will appear in the Yantra 7x Configurator only if the property `yfs.software.tcpip.sockets` defined in `yfs.properties` is set to 'Y'.

The list of attributes that control communication to a weighing scale are:

- ClassName
- PortId
- BaudRate
- DataBits
- StopBits
- Parity
- FlowIn
- FlowOut

Note: The ClassName for the Mettler Toledo Weighing Scale is `com.yantra.ycp.ui.io.YCPToledoPSImpl`. For specifications pertaining to the other attributes, please refer the weighing scale user manual.

14.3.1.1 Setting Up a Device Attribute

Device attributes define the method of communication with the appropriate device. A HP LaserJet printer will have a different parameter list in comparison to a weighing scale. Each individual brand of printer also has its own unique set of parameters and values.

For example, a weighing scale connected through a serial port has specific device attributes including stop bits, parity.

To set up a device attribute:

1. In Device Attributes panel of the Device pop-up window, choose .

2. The Criteria Parameter Details pop-up window appears.
3. Enter information in the applicable fields. Refer [Table 14–4](#) for field value descriptions.
4. Choose .



The image shows a dialog box titled "Criteria Parameter Details". It contains two text input fields: "Parameter Name" and "Parameter Value". At the bottom right, there are two buttons labeled "OK" and "Cancel".

Table 14–4 *Criteria Parameter Details Pop-up Window*

Field	Description
Parameter Name	Enter the parameter name for the device attribute.
Parameter Value	Enter the parameter value for the device attribute.

14.3.2 Creating a New Device from a Device

To create a new device from a device:

1. From the tree in the application rules side panel, choose Device. The Device Setup window appears.
2. In the Device Setup window, select the relevant Device Type and Device Sub Type whose Device is to be copied.
3. The list of Devices is displayed. Select the Device to be copied to create a new device.
4. Choose . The Device pop-up window appears.
5. Enter information in the applicable fields. Refer to [Table 14–2](#) for field value descriptions.
6. Choose .

Yantra 7x supplies a list of standard device type, sub type and individual devices that is supported. The definition of a new device type, sub type and resultant device requires the creation of the appropriate attributes that define the communication with the device.

The list of attributes that control communication to a printer are:

- **DropDirectory** - The directory where the print files are 'dropped' by Yantra 7x Application Server. The Loftware Print Server keeps polling this directory to pick up print requests.
When mentioning the directory structure you can use the full path name or replace the path name with a variable. For more information on including this variable see *Yantra 7x Customization Guide*.

Note: The DropDirectory attribute will appear in the Yantra 7x Configurator only if the property `yfs.ftware.tcpip.sockets` defined in `yfs.properties` is set to 'N'.

- **PrinterAlias** - The printer alias as configured in the Loftware printer setup.

Note: While setting up a Printer device in Yantra 7x, please ensure that the Printer Alias is exactly the same as specified in the Loftware printer set-up.

In instances where a network printer is used, please ensure that the Printer Alias on Yantra 7x does NOT contain the prefix "\\". However, Loftware may require the printer to be defined by prefixing "\\".

- **PrinterServerHostName** - The host name for the Loftware Print Server. While IP Address may be sufficient, the use of host name is recommended for ease of maintenance.
- **PrintServerPort** - The port on which Loftware Print Server listens for print requests. By default, the print server port for Loftware Print Server is 2723.

Note: The PrinterServerHostName and PrintServerPort attributes will appear in the Yantra 7x Configurator only if the property `yfs.loftware.tcpip.sockets` defined in `yfs.properties` is set to 'Y'.

The list of attributes that control communication to a weighing scale are:

- ClassName
- PortId
- BaudRate
- DataBits
- StopBits
- Parity
- FlowIn
- FlowOut

For more information about creation of the appropriate attributes, see [Section 14.3.1.1, "Setting Up a Device Attribute"](#) on page 372.

Note: The ClassName for the Mettler Toledo Weighing Scale is `com.yantra.ycp.ui.io.YCPToledoPSImpl`. For specifications pertaining to the other attributes, please refer the weighing scale user manual.

14.3.3 Modifying a Device

Once a Device has been created, it can be modified.

To modify a device:

1. From the tree in the application rules side panel, choose Device. The Device Setup window appears.
2. In the Device Setup window, select the relevant Device Type and Device Sub Type whose Device is to be modified.
3. The list of Devices is displayed. Select the Device to be modified.

4. Choose . The Device pop-up window appears.
5. Enter information in the applicable fields. Refer to [Table 14–2](#) for field value descriptions.
6. Choose .

14.3.4 Deleting a Device

To delete a device:

1. From the tree in the application rules side panel, choose Device. The Device Setup window appears.
2. In the Device Setup window, select the relevant Device Type and Device Sub Type whose Device is to be deleted.
3. The list of Devices is displayed. Select the Device to be deleted.
4. Choose .

15

Configuring Prints

The operation of a warehouse requires numerous documents, be it labels or reports, to be printed daily. The printing of the documents is either initiated by the occurrence of specific events or is requested ad-hoc by a user.

For example, carrier labels being printed at a manifest station after carton is scanned or a truck manifest (MBOL) being requested when a trailer loading is complete and truck is ready to close.

Documents are printed either individually or in a set or group. A document set consists of multiple documents that are related to individual activity that is performed.

For example, the release of a wave triggers print of wave summary report, carton content labels, batch sheets, and packing slips.

Examples of documents printed in a warehouse include packing lists, BOL, carrier labels, SKU labels, and UCC128 SCM labels.

Yantra 7x provides standard documents that include:

- Batch Sheet for picking
- Cart Manifest for picking
- Packing Slip
- VICS Bill Of Lading (BOL)
- UCC-128 compliant 4x6 Shipping Labels including WALMART® compliance
- UPS Standard carrier labels
- Wave release prints document set consisting of one or more of the above prints

Yantra 7x provides standard documents that include:

A specific document has a label format and device sub type associated to it.

The association of a print document to the device sub type (for example, packing slips on HP LaserJet printers) is done through setting up a device sub type. For more information about setting up a device sub type, see [Section 14.2, "Defining a Device Sub Type"](#) on page 366.

The association of a document to a label format and name is done here.

Use Prints for:

- [Defining Print Documents](#)
- [Defining Label Formats](#)

15.1 Defining Print Documents

A document is assigned a name and a corresponding label format here. Yantra 7x provides standard list of documents for the prints supported.

For example, VICS BOL is associated with the VICS BOL label format.

Print documents and label formats created are at the HUB level.

Use Print Documents set-up for:

- [Creating a Print Document](#)
- [Modifying a Print Document](#)
- [Deleting a Print Document](#)

15.1.1 Creating a Print Document

To create a print document:

1. From the tree in the application rules side panel, choose Prints > Print Documents. The Print Documents window is displayed with Yantra 7x default print documents.
2. Enter information in the applicable fields. Refer [Table 15–1](#) for field value descriptions.
3. Choose .

Print Document	Document Description	Save Directory	Default Label Format
PACKLIST	Packing Slip		PACKLIST
CONTAINER_LABEL	UCC-128 Container Shipping Label		CONTAINER_LABEL_MULTI_SKU
TASKLIST	TaskList		ITEM_PICK_BATCHSHEET
VICS_BOL	VICS Bill Of Lading		VICSBOL_SHIPMENT
UPS_PICKUP_SUMMARY	UPS Pickup Summary		UPS_PICKUP_SUMMARY
UPS_CARRIER_LABEL	UPS Carrier Label		UPS_CARRIER_LABEL
FEDX_CARRIER_LABEL	FEDEX Carrier Label		
COUNTSHEET	Count Sheet		COUNT_SHEET
*			

Results 8 Of 8

Print Documents (DEFAULT)

Table 15–1 Print Documents Window

Field	Description
Print Document	Enter name of the document to be printed.
Document Description	Enter a brief description of the print document.

Table 15–1 Print Documents Window

Field	Description
Save Directory	<p>Enter the directory path where the print document is saved.</p> <p>This is used for documents that are pre-generated but printed on demand at a later time.</p> <p>Typical example is a packing list that is pre-generated, but printed when last carton is scanned.</p> <p>If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i>.</p>
Default Label Format	<p>Choose the default label format for printing.</p> <p>This indicates the default label format for this document across all organizations. Label format is the name of the label design file (.LWL) created using Loftware Label Manager™.</p>

15.1.2 Modifying a Print Document

Once a Print Document has been created, it can be modified.

To modify a print document:

1. From the tree in the application rules side panel, choose Prints > Print Documents. The Print Documents window is displayed with a list of print documents.
2. Enter information in the applicable fields. Refer [Table 15–1](#) for field value descriptions.
3. Choose .

It is recommended that the Yantra 7x provided standard print documents are not modified.

15.1.3 Deleting a Print Document

To delete a print document:

1. From the tree in the application rules side panel, choose Prints > Print Documents. The Print Documents window is displayed with a list of print documents.

2. Choose the Print Document to be deleted.
3. Choose .

It is recommended that the Yantra 7x provided standard print documents are not deleted.

15.2 Defining Label Formats

Label formats corresponding to the documents are defined here. This allows association of a label format to the Loftware™ label format and the mapping XML file.

The Loftware™ label format associated here is created using Loftware™ tools. The mapping XML file is created using *Yantra 7x* supplied toolkit. The field binding between the fields in the label and the field in the standard XML published are specified in the mapping XML.

Refer to *Yantra 7x Installation Guide* for further information on installing and configuring Loftware Label Manager™.

Yantra 7x provides standard label formats and mapping files for all standard documents supported. A print is executed through a service flow defined in the Service Definition Framework (SDF). *Yantra 7x* provides data flow for the standard documents provided.

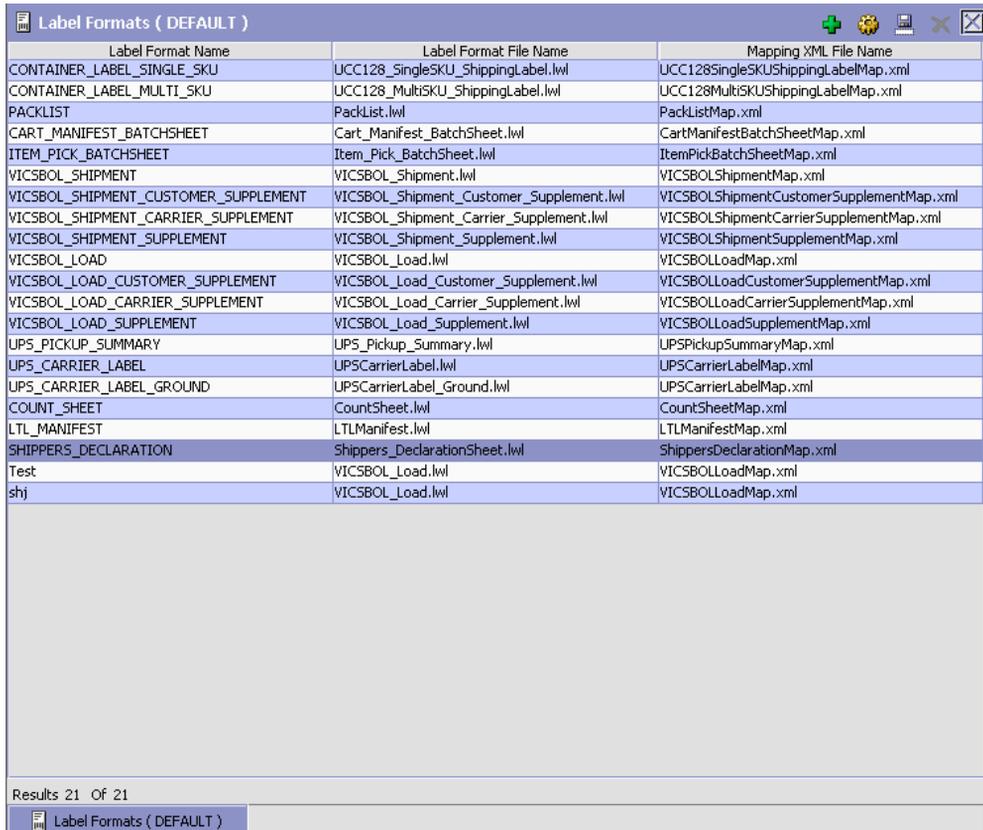
Use Label Format set-up for:

- [Creating a Label Format](#)
- [Modifying a Label Format](#)
- [Deleting a Label Format](#)

15.2.1 Creating a Label Format

To create a label format:

- From the tree in the application rules side panel, choose Prints > Label Formats. The Label Formats window is displayed with *Yantra 7x* default label formats.



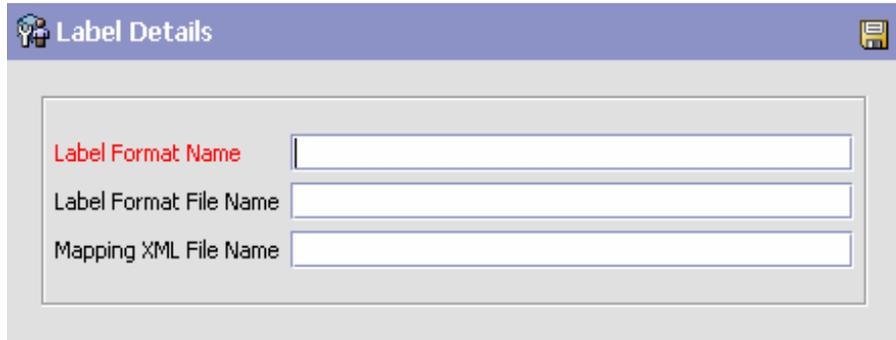
Label Format Name	Label Format File Name	Mapping XML File Name
CONTAINER_LABEL_SINGLE_SKU	UCC128_SingleSKU_ShippingLabel.lwl	UCC128SingleSKUShippingLabelMap.xml
CONTAINER_LABEL_MULTI_SKU	UCC128_MultiSKU_ShippingLabel.lwl	UCC128MultiSKUShippingLabelMap.xml
PACKLIST	PackList.lwl	PackListMap.xml
CART_MANIFEST_BATCHSHEET	Cart_Manifest_BatchSheet.lwl	CartManifestBatchSheetMap.xml
ITEM_PICK_BATCHSHEET	Item_Pick_BatchSheet.lwl	ItemPickBatchSheetMap.xml
VICSBOL_SHIPMENT	VICSBOL_Shipment.lwl	VICSBOLShipmentMap.xml
VICSBOL_SHIPMENT_CUSTOMER_SUPPLEMENT	VICSBOL_Shipment_Customer_Supplement.lwl	VICSBOLShipmentCustomerSupplementMap.xml
VICSBOL_SHIPMENT_CARRIER_SUPPLEMENT	VICSBOL_Shipment_Carrier_Supplement.lwl	VICSBOLShipmentCarrierSupplementMap.xml
VICSBOL_SHIPMENT_SUPPLEMENT	VICSBOL_Shipment_Supplement.lwl	VICSBOLShipmentSupplementMap.xml
VICSBOL_LOAD	VICSBOL_Load.lwl	VICSBOLLoadMap.xml
VICSBOL_LOAD_CUSTOMER_SUPPLEMENT	VICSBOL_Load_Customer_Supplement.lwl	VICSBOLLoadCustomerSupplementMap.xml
VICSBOL_LOAD_CARRIER_SUPPLEMENT	VICSBOL_Load_Carrier_Supplement.lwl	VICSBOLLoadCarrierSupplementMap.xml
VICSBOL_LOAD_SUPPLEMENT	VICSBOL_Load_Supplement.lwl	VICSBOLLoadSupplementMap.xml
UPS_PICKUP_SUMMARY	UPS_Pickup_Summary.lwl	UPSPickupSummaryMap.xml
UPS_CARRIER_LABEL	UPSCarrierLabel.lwl	UPSCarrierLabelMap.xml
UPS_CARRIER_LABEL_GROUND	UPSCarrierLabel_Ground.lwl	UPSCarrierLabelMap.xml
COUNT_SHEET	CountSheet.lwl	CountSheetMap.xml
LTL_MANIFEST	LTLManifest.lwl	LTLManifestMap.xml
SHIPPERS_DECLARATION	Shippers_DeclarationSheet.lwl	ShippersDeclarationMap.xml
Test	VICSBOL_Load.lwl	VICSBOLLoadMap.xml
shj	VICSBOL_Load.lwl	VICSBOLLoadMap.xml

Results 21 Of 21

Label Formats (DEFAULT)

- In the Label Formats window, choose . The Label Details pop-up window displays.
- Enter information in the applicable fields. Refer [Table 15–2](#) for field value descriptions.

4. Choose .



The screenshot shows a 'Label Details' dialog box with a blue header bar containing a key icon on the left and a save icon on the right. The main content area is a light gray rectangle with a thin border, containing three text input fields stacked vertically. The first field is labeled 'Label Format Name' in red text. The second field is labeled 'Label Format File Name' in black text. The third field is labeled 'Mapping XML File Name' in black text.

Table 15–2 Label Details Pop-up Window

Field	Description
Label Format Name	Enter the name of the label format for this label. Label format is the name of the .lwl (Label Design) file created using Loftware Label Manager™.
Label Format File Name	Enter the name of the Loftware™ designed '.LWL' file for this label. For custom labels, enter the file name as extn/ followed by '.lwl' file.
Mapping XML File Name	Enter the file name for the mapping XML for this label. Mapping XML contains the binding or association between the events published XML and the field names used in the label definition. For custom labels, enter the file name as extn/ followed by '.lwl' file.

15.2.2 Modifying a Label Format

Once a Label Format has been created, it can be modified.

To modify a label format:

1. From the tree in the application rules side panel, choose Prints > Label Formats. The Label Formats window is displayed with a list of label formats.

2. Select the Label Format you want to modify and choose .
3. Enter information in the applicable fields. Refer [Table 15–2](#) for field value descriptions.
4. Choose .

It is recommended that Yantra 7x provided standard label formats are not modified.

15.2.3 Deleting a Label Format

To delete a label format:

1. From the tree in the application rules side panel, choose Prints > Label Formats. The Label Formats window is displayed with a list of label formats.
2. Choose the Label Format to be deleted.
3. Choose .

It is recommended that Yantra 7x provided standard label formats are not deleted.

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 463).

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 475).

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 (<=) 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 510).

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 (<=) 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"> • 0001 - Sales Order (Default) • 0002 - Planned Order
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"> It calls <code>beforeConsolidateToShipment</code> in <code>com.yantra.ydm.japi.ue.YDMBeforeConsolidateToShipment</code> for each release before it begins processing. 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>.

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	createDerivedOrder()

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 417.

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 415.

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 (<=) 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 (<=) 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 (<=) 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 427.

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 (<=) 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"> • AUTHORIZATION - Validates the credit card account • CHARGE - Applies the charge to the credit card

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	<code>changeOrder</code>

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
<p>* Note: Some of the elements and attributes are not template driven. Refer to the xml for elements level details.</p>				

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
* Note: 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 (<=) 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>* Note: 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"> • Y - releases line quantities regardless of release date criteria • N - releases lines quantities only after release date criteria have been met. Default.
CheckInventory	Optional. Determine whether inventory should be checked. Valid values are: <ul style="list-style-type: none"> • Y - inventory needs to be checked. Default. • N - inventory does not need to be checked
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 (<=) 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"> • 01 - Optimize on date (Default) • 02 - Optimize on ship node priority • 03 - Optimize on number of shipments
OrderFilter	Optional. Determines the types of orders to filter. Possible values are: <ul style="list-style-type: none"> • A - All orders (Default) • B - Backorders only • N - New orders only
ScheduleAndRelease	Optional. Notify the schedule process to release all releasable line quantities. Valid values are: <ul style="list-style-type: none"> • Y - releases successfully scheduled line quantities. • N - only schedules line quantities. Default. <p>Note: enabling this parameter will not validate hold types configured for the release transaction.</p>

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"> • Y - releases line quantities regardless of release date criteria • N - releases lines quantities only after release date criteria have been met. Default.
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"> • If the OrderFilter is "B" (Backorders Only) and there are no backordered or unscheduled lines. • If the OrderFilter is "N" (New orders Only) and there are some backordered or unscheduled lines. • 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.
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"> • The result of the YFSCheckOrderBeforeProcessingUE user exit returns as false. • The Order has the HoldFlag attribute set to 'Y'. • The Order has the SaleVoided attribute set to 'Y'. • The Order does not have PaymentStatus as AUTHORIZED, INVOICED, PAID, nor NOT_APPLICABLE.
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. Note: 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	sendOrder ()

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 (<=) the current date value in the YFS_Task_Q table.

Events Raised

This transaction raises events specified by the sendOrder () API in the *Yantra Tx 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 (<=) 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 (<=) 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"> • Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. • N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> • Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. • N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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 $EFFECTIVE_TO_DATE \leq (CurrentDate - LeadDays)$

The YFS_RES_POOL_CAPCTY_OVERRIDE table is purged when $CAPACITY_DATE \leq (CurrentDate - LeadDays)$

The YFS_RES_POOL_CAPCTY_CONSMPTN table is purged when $CONSUMPTION_DATE \leq (CurrentDate - LeadDays)$, or if a manual reservation is taken, when $CONSUMPTION_DATE \leq (CurrentDate - Manual\ Reservation\ Lead\ Days)$. When this table is purged, YFS_RES_POOL_CONSMPTN_DTLS will be purged also.

The YFS_RES_POOL_CONSMPTN_DTLS table is purged when $RESERVATION_EXPIRATION_DATE \leq (CurrentDate - LeadDays)$

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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> • Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. • N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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 appropriate section in this 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"> • Y - Default value. Moves qualifying records from the regular tables listed under Table Purged to the corresponding history tables. • N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> • Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. • N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> • Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. • N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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 510.

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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.

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 488.

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"> Y - Default value. Removes qualifying records from the history tables listed under Tables Purged. N- Test mode. Determines the rows that will be removed without actually removing them.
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 485. 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,

Time-Triggered Purge Transactions

- 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"> • Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. • N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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 501.

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"> Y - Default value. Removes qualifying records from the history tables listed under Tables Purged. N- Test mode. Determines the rows that will be removed without actually removing them.
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 499. 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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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 510.

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 401.

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"> Y - Default value. Removes qualifying records from the history tables listed under Tables Purged. N- Test mode. Determines the rows that will be removed without actually removing them.
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 507. 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 401.

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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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	STATTBLPRG
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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 519.

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"> Y - Default value. Removes qualifying records from the history tables listed under Tables Purged. N- Test mode. Determines the rows that will be removed without actually removing them.
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"> Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables. N - Test mode. Determines the rows that will be moved to history tables without actually moving them.
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"> • 1 - current inventory • 0 - inventory within and outside of the ATP time frame. This is the default 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 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 397.

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 (\leq) 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
<p>* These files are located in the following directory: <YFS_HOME>/documentation/api_javadocs/XSD/HTML</p>			

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 397.

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 <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.
InventoryOrganizationCode	Inventory organization code to use when <code>MonitorOption</code> 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 <code>YFS_INVENTORY_ACTIVITY</code> table). 2 – Quick Sync (Re-raise event to publish information in the <code>YFS_INVENTORY_ALERT</code> table). 3 – Full Sync (Monitor based on all inventory items maintained by the inventory organization provided. If no <code>InventoryOrganizationCode</code> 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 data-bbox="763 1067 1149 1119"><AgentCriteriaAttribute> <OverriddenValue></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
<p>* These files are located in the following directory: <YFS_HOME>/documentation/api_javadocs/XSD/HTML</p>			

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

Inventory and Capacity Change Transaction Reference

The following transactions are covered in this appendix:

- [The Inventory Change Transaction](#)
- [The Capacity Change Transaction](#)

B.1 The Inventory Change Transaction

The Inventory Change transaction is used for setting up events that involve inventory changes.

Invoked By

- The `adjustInventory()` API
- Any other API that is capable of doing a status change

Attributes

The following are the attributes for this transaction:

Table B-1 Inventory Change Attributes

Attribute	Value
Base Transaction ID	INVENTORY_CHANGE
Base Document Type	General
Base Process Type	General

Table B-1 Inventory Change Attributes

Attribute	Value
Abstract Transaction	No
APIs Called	None

Events Raised

This transaction raises the following events:

- **SUPPLY_CHANGE**

This event is raised for every inventory supply change within Yantra 7x. It publishes detailed information about the changed supply.

Key Data: YFS_HOME/documentation/api_javadocs/dbd/INVENTORY_CHANGE_SUPPLY_CHANGE_dbd.txt

Data Published: YFS_HOME/documentation/api_javadocs/XSD/HTML/INV_INVENTORY_CHANGE_SUPPLY_CHANGE.html

- **DEMAND_CHANGE**

This event is raised for every inventory demand change within Yantra 7x. It publishes detailed information about the changed demand. Order information will be published when the inventory organization is set to create demand details.

Key Data: YFS_HOME/documentation/api_javadocs/dbd/INVENTORY_CHANGE_DEMAND_CHANGE_dbd.txt

Data Published: YFS_HOME/documentation/api_javadocs/XSD/HTML/INV_INVENTORY_CHANGE_DEMAND_CHANGE.html

- **EXTERNAL_SUPPLY_CHANGE**

This event is raised for every supply change for externally maintained inventories within Yantra 7x.

Key Data: YFS_HOME/documentation/api_javadocs/dbd/INVENTORY_CHANGE_EXTERNAL_SUPPLY_CHANGE_dbd.txt

Data Published: YFS_HOME/documentation/api_javadocs/XSD/HTML/INV_INVENTORY_CHANGE_EXTERNAL_SUPPLY_CHANGE.html

- EXTERNAL_DEMAND_CHANGE

This event is raised for every demand change for each organization that does not maintain its inventory within Yantra 7x.

Key Data: YFS_HOME/documentation/api_javadocs/dbd/INVENTORY_CHANGE_EXTERNAL_DEMAND_CHANGE_dbd.txt

Data Published: YFS_HOME\documentation\api_javadocs\XSD\HTML\INV_INVENTORY_CHANGE_EXTERNAL_DEMAND_CHANGE.html

- INVENTORY_CHANGE

This event is raised for each inventory supply change within Yantra 7x. It publishes detailed information of the changed supply.

Key Data: YFS_HOME/documentation/api_javadocs/dbd/INVENTORY_CHANGE_INVENTORY_CHANGE_dbd.txt

Data Published: YFS_HOME/documentation/api_javadocs/dbd/INVENTORY_CHANGE_INVENTORY_CHANGE_dbd.txt

Note: When defining one of the following events:

- SUPPLY_CHANGE
- DEMAND_CHANGE
- EXTERNAL_SUPPLY_CHANGE
- EXTERNAL_DEMAND_CHANGE

You should not call the `changeOrder` API as a result of these event. Doing so could cause the API to raise the event that called it, creating an infinite cycle.

B.2 The Capacity Change Transaction

The Capacity Change transaction is used for setting up events that involve capacity changes.

Invoked By

- The `adjustInventory()` API
- The `createResourcePool()` API

- The `modifyResourcePool()` API
- The `overrideResourcePoolCapacity()` API
- Any other API that is capable of allocating capacity against external resource pool

Attributes

The following are the attributes for this transaction:

Table B–2 Capacity Change Attributes

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

Events Raised

This transaction raises the following events:

- ON_CAPACITY_ADJUSTMENT

This event is raised for every standard capacity adjustment when the `createResourcePool()` API or `modifyResourcePool()` APIs are invoked. It publishes detailed information about the standard capacity adjustments.

Key Data: `YFS_HOME/documentation/api_javadocs/dbd/CAPACITY_CHANGE_ON_CAPACITY_ADJUSTMENT_dbd.txt`

Data Published: `YFS_HOME/documentation/api_javadocs/XSD/HTML/INV_CAPACITY_CHANGE.ON_CAPACITY_ADJUSTMENT.xml`

- ON_OVERRIDE_CAPACITY

This event is raised for every adjustment of overridden capacity when the `overrideResourcePoolCapacity()` API is invoked. It publishes detailed information about the changes in the overridden capacity.

Key Data: YFS_HOME/documentation/api_javadocs/dbd/ CAPACITY_CHANGE_ON_OVERRIDE_CAPACITY_dbd.txt

Data Published: YFS_HOME/documentation/api_javadocs/XSD/HTML/ INV_CAPACITY_CHANGE.ON_OVERRIDE_CAPACITY.xml

- EXTERNAL_CAPACITY_CHANGE

This event is raised for every external capacity change within Yantra 7x. It publishes detailed information about the changed capacity of external resource pools.

Key Data: YFS_HOME/documentation/api_javadocs/dbd/ CAPACITY_CHANGE_EXTERNAL_CAPACITY_CHANGE_dbd.txt

Data Published: YFS_HOME/documentation/api_javadocs/XSD/HTML/ YFS_CAPACITY_CHANGE.EXTERNAL_CAPACITY_CHANGE.xml

Note: When defining the EXTERNAL_CAPACITY_CHANGE event, you should not call the changeOrder API as a result of this event. Doing so could cause the API to raise the event that called it, creating an infinite cycle.

Service Builder Nodes and Parameters

This appendix provides a complete list of the transport, component and adapter nodes used in the Service Builder.

C.1 Transport Nodes

Transport nodes determine how data should be transferred from one location to another. To define their configuration properties, click the links connection going into or out of the transport node. The link going *into* a transport node is a sender link and the link going *out* is a receiver link.

For detailed information about each type of transport node, see the following topics:

- [Database](#)
- [DCS 6.2 Database](#)
- [Component Object Model \(COM\)](#)
- [Enterprise Java Beans \(EJB\)](#)
- [File Input/Output \(File I/O\)](#)
- [File Transfer Protocol \(FTP\)](#)
- [Hypertext Transport Protocol \(HTTP\)](#)
- [WebService](#)
- [Synchronous WebLogic and MQSeries](#)
- [WebLogic and MQSeries JMS Queue](#)
- [Microsoft Message Queue \(MSMQ\)](#)

Exception Handling

Asynchronous receiver links can be configured to handle exceptions, as described in [Section C.1.12, "Receiver Link Exception Handling"](#) on page C-607. The asynchronous receiver links that can be configured for exception handling are the following:

- [Database Receiver](#)
- [File I/O Receiver](#)
- [FTP Receiver](#)
- [WebLogic and MQSeries JMS Receiver](#)
- [MSMQ Receiver](#)

C.1.1 Database

The database node defines an asynchronous transport using a database table. The database transport node can have sender and receiver links, the links capture the respective properties.

Note: Yantra recommends that you do not use the database node when scaling. Instead, we recommend that you use a JMS Queue node instead.

C.1.1.1 Database Sender

Configuration Properties

The following are the properties of the links connecting this node:

Table C-1 Database Sender Configuration Properties

Property	Description
Runtime Tab	
Table Name	Select the table you want the message to be written to. Valid values are YFS_IMPORT and YFS_EXPORT.

Table C-1 Database Sender Configuration Properties

Property	Description
Rollback on Exception	<p>Select this check box if you want the message to be committed to the database only after the service is completed.</p> <p>Uncheck this check box if you want the message to be committed to the database immediately.</p> <p>For example, if the ON_SUCCESS event of any standard Yantra 7x API is attached to a service, in which the message is transactionally written to the database, the message is committed to the database only upon successful completion of the ON_SUCCESS event. The message is then rolled back from the queue if there is any error in the ON_SUCCESS event after the message is staged. However, in non-transactional mode, the message remains in the database, once it is staged and is not rolled back.</p>
Header Tab	
Header Name	<p>The Header Name and Header Value allow the sender to differentiate between messages in the table. See the Selector field in the "Database Receiver Configuration Properties" table.</p> <p>Choose  to add a new header name and value.</p> <p>Choose  to modify an existing header name and value.</p> <p>Choose  to delete an existing header name and value.</p> <p>Important: Do not enter any spaces in the Header Name.</p>

Table C–1 Database Sender Configuration Properties

Property	Description
Header Value	<p>Saved in the USER_REFERENCE column of the YFS_EXPORT and YFS_IMPORT tables as a name value pair. A maximum limit of these name/value pairs stored is 2048 bytes, beyond which the references are truncated.</p> <p>Each Header Name/Value pair specified is appended in the USER_REFERENCE column of the YFS_IMPORT/YFS_EXPORT table as <i>Name1=Value1 Name2=Value2</i></p> <p>These references can be used to identify the key data stored in the MESSAGE field of the YFS_IMPORT/YFS_EXPORT table when querying.</p> <p>This can be set to dynamically extract from the message using the following syntax <code>xml://<full path of the element from root node>/@<attribute name></code>.</p> <p>For example, to get the sales order number from the publish ship advice output XML: set the value as <code>xml://ShipmentAdvices/ShipmentAdvice/@SalesOrderNo</code> results in the USER_REFERENCE field to be populated with <code>NAME1=<value of attribute SalesOrderNo in the XML></code></p> <p>Important: Do not enter any spaces in the Header Value.</p>

C.1.1.2 Database Receiver

Configuration Properties

The following are the properties of the links connecting this node:

Table C–2 Database Receiver Configuration Properties

Property	Description
Runtime Tab	
Sub Service Name	Enter a unique identifier for each asynchronous receiver.

Table C–2 Database Receiver Configuration Properties

Property	Description
Initial Threads	<p>Enter the number of threads that can process messages simultaneously. This value must be a minimum of 1. Based on your throughput requirements, you can increase the number of threads.</p> <p>The number of threads set in the configuration can be increased for a Sub Service Name dynamically from the System Management Console. For more information about the System Management Console, see the <i>Yantra 7x Platform User Guide</i>.</p>
Selector	<p>Enter the message to be processed, using the Header Name and Header values from the "Database Sender Configuration Properties" table, implemented as an SQL where clause in the of form <code>USER_REFERENCE='<name>=value'</code></p> <p>Note: When specifying a selector, use only single quotes.</p> <p>Note: If you configure two services to read from the same table (and select the same records), the results may be unpredictable. For a database, the columns that can be used are <code>FLOW_NAME</code>, <code>SUB_FLOW_NAME</code>, <code>USER_REFERENCE</code>, from the <code>YFS_EXPORT/IMPORT</code> table.</p>
Table Name	<p>Select the table you want the message to be read from. Valid values are <code>YFS_IMPORT</code> and <code>YFS_EXPORT</code>.</p> <p>Must match the table specified in the receiver link.</p>
Polling Frequency (seconds)	<p>Enter the frequency in seconds to poll for messages from the database table. Defaults to 600 seconds (10 minutes).</p> <p>A separate thread manages all exceptions resolved from the exception console and polls the database every 60 seconds (1 minute). The frequency of polling for exception processing cannot be modified.</p>
Service to Execute on EOF Message	<p>Required if the message contains an EOF message ID.</p> <p>Choose  to select the service to be invoked when an EOF message is received. Once the EOF message is received, the framework waits for a few minutes (configurable in the <code>yfs.properties</code>) before executing this service. For more information see, Section C.1.5.3, "Enabling EOF Messages in the Platform Framework".</p>

Table C–2 Database Receiver Configuration Properties

Property	Description
Root Node Name of EOF Message	This need to be specified only if the message contains an EOF message ID. Enter your custom root node name for the EOF message. By default the end of file message has a root node name as "EOF". For more information see, Section C.1.5.3, "Enabling EOF Messages in the Platform Framework"
Server Tab	
Server Name	Required. Name of the integration server instance which actually executes the service. For more information on creating a new server refer to " Adding a New Server ".
Exception Tabs	See the " Receiver Link Exception Handling Properties " table.

Connection Properties

The following are the Database node's connection properties:

Table C–3 Database Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Yes, for services invoked both in a synchronous or asynchronous mode
Can be placed before	<ul style="list-style-type: none"> Any component node Any transport node (except for FTP, JMS, or File I/O); use a Pass-through node to connect them
Can be placed after	<ul style="list-style-type: none"> Start node Any synchronous transport node. Any other component node Any asynchronous transport node (except for FTP, JMS, or File I/O); use a Pass-through node to connect them
Passes data unchanged	Transport nodes do not modify data

C.1.2 DCS 6.2 Database

The Distribution Center System (DCS) 6.2 database node defines an asynchronous transport using a DCS 6.2 database table. The DCS 6.2 database transport node enables you to define services or interfaces to read from or write to the DCS 6.2 tables that are comprised of positional data. The interface tables in the DCS 6.2 database are `INFC_UPLD_TAB_1` and `INFC_DNLD_TAB_1`. Multiple records in these interface tables constitute a transaction.

You can define defaults which can be applied on these interface tables. Hence, if DCS 6.2 needed some specific attributes to be set in the data and that data is not available in the external system, interface defaulting mechanisms can be used to set them up. For more information on using the defaulting component refer to [Section C.2.10, "Defaulting Component"](#).

The DCS 6.2 database component reads unprocessed interface table (`INFC_DNLD_TAB_1`) records for an interface type and node. If any of the components following the DCS 6.2 database component in the service definition framework throws an exception, the interface records for that transaction are marked as error. Processing of these error records is based on a system property that is passed to a server JVM running the inbound interface.

Note: If a text translator component converts the list of interface records into an XML, an XSL transformation must be applied on the output of the text translator to conform the XML to the input required by the API.

C.1.2.1 DCS 6.2 Database Sender

Configuration Properties

The following are the properties of the links connecting this node:

Table C-4 DCS 6.2 Database Sender Configuration Properties

Property	Description
Interface Type	Enter the interface type. The records of this interface type are processed.
Table Type	Select either UPLD or DNLD for the upload or download interface tables. The interface table names are derived from this and the value entered in the Node field. The default value for the sender is UPLD.
Node	Enter the ship node associated with the service.
Table Name	This field is not editable. The table name is automatically entered once the Table Type is selected and is appended with the entered ship node. The table name is YFS_INFC_UPLD_TAB1_Node1 for UPLD and YFS_INFC_DNLD_TAB1_Node1 for DNLD table types with the ship node as Node1.
Remote Host ID	Enter the remote host ID in the database.
Rollback on Exception	Check this box if you want to rollback the database write if the parent transaction fails. Uncheck this check box if you want the message to be committed to the database immediately. For example, if the ON_SUCCESS event of any standard Yantra 7x API is attached to a service in which the message is transactionally written to the database, the message is committed to the database only upon successful completion of the ON_SUCCESS event. The message is then rolled back from the database if there is any error in the ON_SUCCESS event after the message is staged. However, in non-transactional mode, the message remains in the database, once it is staged and is not rolled back.

C.1.2.2 DCS 6.2 Database Receiver

Configuration Properties

The following are the properties of the links connecting this node:

Table C-5 DCS 6.2 Database Receiver Configuration Properties

Property	Description
Runtime Tab	

Table C-5 DCS 6.2 Database Receiver Configuration Properties

Property	Description
Sub Service Name	Enter a unique identifier for each asynchronous receiver.
Node	Enter the ship node associated with the service.
Selector	Optional. This field can be used for selecting a specific remote host ID. Note: When specifying a selector, use only single quotes.
Table Type	Select either UPLD or DNLD for the upload or download interface tables. The interface table names are derived from this and the value entered in the Node field. The default value for the receiver is DNLD.
Table Name	This field is not editable. The table name is automatically entered once the Table Type is selected and is appended with the entered ship node. The table name is YFS_INFC_UPLD_TAB1_Node1 for UPLD and YFS_INFC_DNLD_TAB1_Node1 for DNLD table types with the ship node as Node1.
Interface Type	The interface type in which the records are processed.
Polling Frequency (seconds)	Enter the frequency in seconds to poll for messages from the database table. Defaults to 600 seconds (10 minutes).
Server Tab	
Server Name	Required. Name of the integration server instance which actually executes the service. For more information on creating a new server refer to "Adding a New Server" .
Exception Tabs	See the "Receiver Link Exception Handling Properties" table.

Connection Properties

The following are the DCS 6.2 Database node's connection properties:

Table C–6 DCS 6.2 Database Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Yes, for services invoked both in synchronous or asynchronous mode.
Can be placed before	<ul style="list-style-type: none"> Text Translator component node End node
Can be placed after	<ul style="list-style-type: none"> Start node Text Translator component node
Passes data unchanged	Transport nodes do not modify data

C.1.3 Component Object Model (COM)

The Component Object Model (COM) transport Node defines the synchronous COM call being made to the configured COM component.

Note: Make sure the <YFS_HOME>/bin directory is in your system path.

Configuration Properties

The following are the properties of this node:

Table C–7 COM Sender Configuration Properties

Property	Description
Program ID	<p>Enter the COM component’s program ID to be invoked. The program ID can be found in the registry by running the regedit utility and looking up the DLL to be invoked.</p> <p>The custom COM component should implement an execute method, the method signature should be as follows:</p> <pre>[id(1), helpstring("method execute")] HRESULT execute([in]BSTR sData, [out]VARIANT *outData, [out, retval] long *RetVal);</pre>

Connection Properties

The following are the COM node’s connection properties:

Table C–8 COM Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any synchronous transport node Any other component node Any asynchronous transport node (except for FTP or File I/O); use a Pass-through node to connect them.
Passes data unchanged	No. The data returned by the COM component is passed to the next service component.

C.1.4 Enterprise Java Beans (EJB)

The Enterprise Java Beans (EJB) Transport node defines the way messages are sent synchronously using the EJB protocol. The EJB transport node has sender-related properties.

Note: The EJB node is only needed to call non-Yantra EJBs.

Configuration Properties

The following are the properties of this node:

Table C–9 EJB Sender Configuration Properties

Property	Description
Provider URL	Enter the provider URL for the JNDI lookup of the EJB home. <ul style="list-style-type: none"> For WebLogic, set to t3://<DNS Server Name or IP Address>:<port>
EJB Home Name	Enter the class name implementing the EJB home interface. This is used by the client to find, create, or remove EJB objects.
JNDI Name	Enter the name used to lookup the EJB home interface.

Table C–9 EJB Sender Configuration Properties

Property	Description
Initial Context Factory	The class name of the initial context factory. This is the class used for resolution of names for naming and directory operations.
Method Name	Enter the method to be invoked. This method must have a standard signature: Document outdoc = method(Document indoc)
Authenticate	If selected, security-related authentications can be defined for the security principal and security credential. The Security Principal and Credentials are required for the JNDI lookup.
Security Principal	If you selected Authenticate, enter the user name for the Access control list of the EJB container.
Security Credential	If you selected Authenticate, enter the password for the Access control list of the EJB container.

Connection Properties

The following are the EJB node's connection properties:

Table C–10 EJB Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> • Any transport node except FTP or File I/O • Any other component node
Can be placed after	<ul style="list-style-type: none"> • Start node • Any synchronous transport node • Any other component node • Any asynchronous transport node (except for FTP or File I/O); use a Pass-through node to connect them.
Data may be changed	Data may be changed depending on what method is called by the transport node.

C.1.5 File Input/Output (File I/O)

The File Input/Output transport node defines how messages are exchanged asynchronously using flat files. Files can be created or processed using this flow component. It allows you to configure how to create or process files asynchronously.

C.1.5.1 File I/O Sender

Configuration Properties

The following are the properties of links connecting this node:

Table C–11 File I/O Sender Connection Properties

Property	Description
Working Directory	Enter the directory into which the incoming files are placed before processing begins. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Completion Directory	Enter the directory into which a file is moved after it has been successfully processed or after it exceeds the sizes specified by Max Output File Size or Max Transactions Per File. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
File Prefix	Enter the name to prefix to each file name. Used to differentiate output files from different services.
File Suffix	Enter the name to append to each file name. Used to differentiate output files from different services.
Max Output File Size (MB)	Enter the maximum size (in MB) that can be reached before closing the current file and opening a new one. Starts writing to a new file when the minimum of Max Out File Size and Max Transactions Per File is exceeded.

Table C–11 File I/O Sender Connection Properties

Property	Description
Max Transactions Per File	Enter the maximum number of messages to write to a file before closing the current file and opening a new one. Starts writing to a new file when the minimum of Max Out File Size and Max Transactions Per File is exceeded.
EncodingType	Enter the character encoding of the input file. Takes JVM-supported values such as UTF-8, UTF-16, ISO-8859. Defaults to the platform default for the JVM unless otherwise specified.
Max Time to Rollover (mins)	Enter the maximum time (in minutes) the output file remains open.

C.1.5.2 File I/O Receiver

Configuration Properties

The following are the properties of this node:

Table C–12 File I/O Receiver Configuration Properties

Property	Description
Runtime Tab	
Sub Service Name	Enter a unique identifier for each asynchronous receiver within a service definition. Note: When configuring a sub-flow, do not enter a Sub Service Name beginning with "YIF".
Includes Pattern	Enter the comma separated list of files to process together. For example, to specify all files with the posCreateOrder.* name, use .*\ <code>posCreateOrder.*</code> . If not specified, processes all files in the incoming directory in the order specified by the File Processing Sequence.
Encoding Type	Enter the character encoding of the input file. Takes JVM-supported values such as UTF-8, ASCII, ANSI. Defaults to the platform default for the JVM, unless otherwise specified.
File Processing Sequence	Enter the order in which to processes files from the Incoming Directory. Values are: LastModifiedTime (date) and ByName (name). Defaults to Last Modified Time.

Table C–12 File I/O Receiver Configuration Properties

Property	Description
Maximum Errors Per File	Enter the number of logged errors before the Receiver to terminates processing the file. If set to 0, it defaults to 1. If not specified, defaults to 10.
Polling Frequency (seconds)	Enter the time (in seconds) to poll the source directory for unprocessed files. Defaults to 600 seconds.
Create EOF Message	Select this field if you want to create End of File (EOF) message. If checked the framework creates an EOF message at the end of each file processed and pass to the next service component. For more information on creating EOF messages see, Section C.1.5.3, "Enabling EOF Messages in the Platform Framework" .
File Tab	
Incoming Directory	Enter the directory in which to look for input files. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Completion Directory	Enter the directory into which the processed files are archived after they have been processed. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Working Directory	Enter the directory into which the files from the incoming directory are moved before processing begins. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .

Table C–12 File I/O Receiver Configuration Properties

Property	Description
Error Directory	Enter the directory into which any errors in the file being processed are created. A new error file is created for each file processed. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Error File Suffix	Enter the name to append to error files. The error file records errors that occur while processing the input file. Defaults to <code>.err</code> .
Completion File Suffix	Required if the Completion Directory is specified. Defaults to <code>.done</code> .
Server Tab	
Server Name	Enter the name of the integration server instance which actually executes the service. For more information on creating a new server refer to " Adding a New Server ".
Exception Tabs	See the " Receiver Link Exception Handling Properties " table.

Connection Properties

The following are the File I/O node's connection properties:

Table C–13 File I/O Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked asynchronously
Can be placed before	<ul style="list-style-type: none"> • FTP transport node • Text Translator component node • End node
Can be placed after	<ul style="list-style-type: none"> • Start node • FTP transport node • Text Translator component node
Passes data unchanged	Transport nodes do not modify data

C.1.5.3 Enabling EOF Messages in the Platform Framework

The creation of EOF messages is divided into two parts:

1. [Modifying File Input/Output Processing](#)
2. [Modifying the JMS Queue and Database Receiver to Handle EOF Messages](#)

C.1.5.3.1 Modifying File Input/Output Processing

The File I/O receiver stamps the outgoing XML messages with a unique message group identification, `YantraMessageGroupID` before it is sent to the next component in the service definition framework. Once the end of the file is reached, an EOF message is created with the same message group ID. This EOF message is useful if special processing needs to be done at the end of each file.

Rules for creating EOF messages:

If you are using Yantra 7x File IO adapter component then the same unique message group id is added and the message header is appended to each message by default.

The following are the rules for creating EOF messages when using a third party component and integrating with Yantra 7x:

- The EOF message must be of the format:

```
<EOF YantraMessageGroupID="Mandatory" />
```

- The XML root node name must be `EOF` and the `YantraMessageGroupId` is a required attribute. This attribute is essential in identifying all messages that belong to a group.
- When Yantra is writing the EOF messages to a JMS queue or a Database, then a message header with `MessageType="EOF"` is created by the framework.

Note: If you are using a third party component for writing EOF messages into JMS queues or Database, then you should make sure that the EOF message has a header of `MessageType="EOF"`.

Example scenario for creating EOF messages

The steps involved in creating EOF messages and processing the messages in a JMS queue or a database is explained in detail with a sample XML file as an input to the File I/O adapter.

The following figure shows the service framework with a File IO component.



1. The input file can be a delimited file, text file or an XML file. In this example we consider an XML file.

Example C-1 Sample Input File from File I/O Node

```
<Items Attr1="Value1" Attr2="Value2" Attr3="Value3" >
  <Item ItemId="Item1" />
  <Item ItemId="Item2" />
  <Item ItemId="Item3" />
</Items>
```

2. The parsed input XML file of each child node is then passed into the JMS queue or a Database as a separate message.
3. The framework then appends all the input XML files root node attributes to each of the message put in the queue.

Note: If the input XML's root node do not contain a `YantraMessageGroupID`, the framework generates a unique ID and append to each message put into the queue.

If the input files are non-XML files then the root node attributes does not get included in the EOF node. It would contain the attributes given below:

```
<EOF YantraMessageGroupID="file1.txt.001" FileName=""
FileSize="" LastModifiedTime="" />
```

4. The EOF element contains the attributes in the root element of the input XML file along with the file name, file size (in bytes) and the last modified time of the file.

C.1.5.3.2 Modifying the JMS Queue and Database Receiver to Handle EOF Messages

The messages received at the queue or the database is explained with the example as described in [Example C–2](#).



Example C–2 Output Messages created in the JMS queue or Database

1. `<Item YantraMessageGroupID="file1.txt.001" ItemId="Item1" Attr1="Value1" Attr2="Value2" Attr3="Value3" />`
2. `<Item YantraMessageGroupID="file1.txt.001" ItemId="Item2" Attr1="Value1" Attr2="Value2" Attr3="Value3" />`
3. `<Item YantraMessageGroupID="file1.txt.001" ItemId="Item3" Attr1="Value1" Attr2="Value2" Attr3="Value3" />`

```
</>
```

```
4. <EOF YantraMessageGroupID="file1.txt.001" ItemId="Item4"  
Attr1="Value1" Attr2="Value2" Attr3="Value3" FileName=""  
FileSize="" LastModifiedTime="" />
```

1. Once the EOF is reached the messages received by the JMS queue or Database the framework checks whether the EOF has the same message group ID.
2. You must configure the JMS queue or Database receiver component to enable a service to be invoked when the EOF message is received. If a service is not invoked then an exception is thrown.
3. The framework is set wait for a certain amount of time which can be configured in `yfs.properties` file, before calling the service.
4. In general, the XML file passed to the service must have a root node name of `EOF`. However you can specify a custom root node name instead of `EOF`. This node name is specified in the JMS queue or Database receiver properties.

Error Handling by Integration Server:

When an EOF file message reaches the integration server it is checked for any reprocessable messages for this service with the same `YantraMessageGroupID`. If there are any pending error messages to be reprocessed, then the EOF messages are marked as reprocessable. This error messages are then inserted into the `YFS_REPROCESS_TABLE`.

C.1.6 File Transfer Protocol (FTP)

The FTP node allows for sending and receiving files. Files residing in a local directory are sent to a remote directory on an FTP server. Files residing in a remote directory are received from an FTP server and stored in a local directory.

Note: Ensure that all source and destination directories and files have read/write permissions for the remote user specified and for the user running the Integration Server.

Note: The FTP server is not multi-threaded.

C.1.6.1 FTP Sender

Configuration Properties

The following are the properties of this node:

Table C–14 FTP Sender Configuration Properties

Attribute	Description
Runtime Tab	
Sub Service Name	Enter a unique identifier for each asynchronous receiver within a service definition.
Server Address	Enter the IP address of the remote FTP server.
Port	Enter the port number of the remote FTP server. Default is port 21.
User ID	Enter the login ID to use on the FTP server.
Polling Frequency	Enter the time in seconds to poll the source directory for unprocessed files.
Includes Pattern	Enter the regular expression pattern of the files to retrieve. If not specified all files in the working directory are included. The following are examples of regular Perl expressions that can be used: <ul style="list-style-type: none"> • *.*.txt - All text files. • temp.*.txt - All text files containing the pattern of 'temp'.
Password	Enter the password to use on the FTP server
Binary Transfer	Select this option to transfer data that contains non-ascii characters.
ASCII Transfer	Select this option to transfer text files.

Table C–14 FTP Sender Configuration Properties

Attribute	Description
Compress File	Select this field to compress and deliver a zip file to the remote FTP server.
Local Zip Completion Dir	If Compressed File is selected, enter the directory in which the compressed files is to be stored. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Source File Parameters Tab	
Working Directory	Enter the directory on the FTP server from which files are transferred. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Completion Directory	Enter the remote completion directory files are to be moved to after they are successfully transferred from the remote staging directory. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
File Separator	Enter the file separator used on the FTP server. This field is mandatory.
Completion Suffix	Enter the suffix to append the file when the file from the remote working directory is moved into the remote completion directory.
Destination File Properties Tab	
Working Directory	Enter the directory in which the files received are transferred to. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .

Table C–14 FTP Sender Configuration Properties

Attribute	Description
Completion Directory	Enter the local completion directory files are to be moved to after they are successfully transferred from the remote working directory. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
File Separator	Enter the file separator used on the FTP server. This field is mandatory.
Completion Suffix	Enter the suffix to append the file when the file from the local working directory is moved into the local completion directory.
Server Tab	
Server Name	Enter the name of the server which actually executes the service. For more information on creating a new server refer to " Adding a New Server ".
Exception Tabs	See the " Receiver Link Exception Handling Properties ". table

Important: Do not configure multiple instances of file receivers or senders trying to FTP the same file, this can lead to file data loss.

C.1.6.2 FTP Receiver

Configuration Properties

The following are the properties of this node:

Table C–15 FTP Receiver Configuration Properties

Attribute	Description
Runtime Tab	
Sub Service Name	Enter a unique identifier for each asynchronous receiver within a service definition.

Table C–15 FTP Receiver Configuration Properties

Attribute	Description
Server Address	Enter the IP address or host name of the remote FTP server.
Port	Enter the port number of the remote FTP server. Defaults to port 21.
User ID	Enter the login ID to use on the FTP server.
Polling Frequency	Enter the time in seconds to poll the source directory for unprocessed files.
Includes Pattern	<p>Enter the regular expression pattern of the files to retrieve. If not specified all files are included in the working directory.</p> <p>The following are examples of regular Perl expressions that can be used:</p> <ul style="list-style-type: none"> • *.*.txt - All text files. • temp.*.txt - All text files containing the pattern of 'temp'.
Password	Enter the password to use on the FTP server.
Binary Transfer	Select this option to transfer data that contains non-ascii characters.
ASCII Transfer	Select this option to transfer text files.
Decompress File	Select this field if the file being received is compressed.
Source File Parameters Tab	
Working Directory	<p>Enter the directory on the FTP server from which files are to be transferred.</p> <p>Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i>.</p>
Completion Directory	<p>Enter the completion directory. After successfully transferred from the remote staging directory, the file is moved on the server into the completion directory.</p> <p>Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i>.</p>

Table C–15 FTP Receiver Configuration Properties

Attribute	Description
File Separator	Enter the file separator used on the FTP server. This field is mandatory.
Completion Suffix	Enter the suffix to append the file when the file from the remote working directory is moved into the remote completion directory.
Destination File Properties Tab	
Working Directory	Enter the directory in which the files received are transferred to. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Completion Directory	Enter the completion directory. After a file is transferred successfully from the remote working directory, it is subsequently moved to the local completion directory. Note: Ensure the directory has the appropriate read/write permissions and that the full path name has been specified. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
File Separator	Enter the file separator used on the FTP server. This field is mandatory.
Completion Suffix	Optional. The suffix to append the file when the file from the local working directory is moved into the local completion directory.
Server Tab	
Server Name	Enter the name of the server which actually executes the service. For more information on creating a new server refer to "Adding a New Server" .
Exception Tabs	See the "Receiver Link Exception Handling Properties" table.

Important: Do not configure multiple instances of file receivers or senders trying to FTP the same file, this can lead to file data loss.

Connection Properties

The following are the FTP node's connection properties:

Table C–16 FTP Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked asynchronously
Can be placed before	<ul style="list-style-type: none"> • File I/O transport node • End node • Pass-through cannot follow FTP
Can be placed after	<ul style="list-style-type: none"> • Start node, only for asynchronously invoked services • File I/O transport node. • Pass-through cannot precede FTP
Passes data unchanged	Transport nodes do not modify data

C.1.7 Hypertext Transport Protocol (HTTP)

The HTTP transport node defines the way synchronous messages are sent using the HTTP post method. The HTTP transport node has sender-related properties.

Configuration Properties

The following are the properties of this node:

Table C–17 HTTP Configuration Properties

Property	Description
URL	Enter the URL to which the message is to be posted.
HTTP Post Variable	Enter the variable name to which the HTTP post data is to be assigned.

Table C–17 HTTP Configuration Properties

Property	Description
Is Secure	If this field is selected, the message is encrypted when being posted to the URL specified.
Key Store Type	If Is Secure is checked, set this value to JKS (Java Key Store).
Key Store	If Is Secure is selected, enter the key store for storing client side digital certificates. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Key Store Passwd	If Is Secure is selected, enter the password to access the key store.
Trust Store	If Is Secure is selected, enter the trust store for storing server side digital certificates. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Trust Store Passwd	If Is Secure is selected, enter the password to access the trust store.

Note: Making a secure HTTP call requires modifying the <JAVA_HOME>/jre/lib/security/java.security file. Comment the following line from the file:

```
security.provider.2=com.sun.rsajca.Provider
```

Commenting the property enables the runtime loading of the security.provider variable set later in the code. If you do not want to change the common Java Security file, refer to BEA documentation about using the weblogic.policy file and modify it accordingly. Use the weblogic.policy file with the necessary changes as suggested on the BEA web site at:

<http://e-docs.bea.com/wls/docs81/secintro/concepts.html#1091743>

You can enable the Java Security Manager to use the weblogic.policy security policy file by adding these parameters to the WebLogic startup script:

```
java -Djava.security.manager
-Djava.security.policy=<WLS_HOME>/lib/weblogic.policy
```

Connection Properties

The following are the HTTP node's connection properties:

Table C-18 HTTP Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except for FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any synchronous transport node Any other component node Any asynchronous transport node (except for FTP or File I/O); use a Pass-through node to connect them
Passes data unchanged	Transport nodes do not modify data

C.1.8 WebService

This WebService transport node allows the Service Definition Framework to make outbound webservice calls. For more information about WebServices and how to create yantra.ear, see the *Yantra 7x Installation Guide*.

Important: WebServices cannot be used to call APIs in backward compatibility mode. Also, outbound WebService calls using the Service Builder do not support HTTPS protocol.

Configuration Properties

The following are the properties of this node:

Table C–19 *WebServices Configuration Properties*

Property	Description
General Tab	
URL	Enter the URL to which the message is to be posted.
Target Object URN	Enter the web service's resource name.
Is Yantra Webservice	Check this box to indicate that this service is calling a Yantra 7x Webservice. If this checkbox is selected, the parameter name defaults to <code>apiString</code> and the Parameter Name text box on this tab is disabled.
Parameter Name	Enter the name of the document parameter.
Encoding Style URI	Enter the name of the encoding you want to use.
Method Name	Enter the name of the method you want to invoke.
Is Secure	If this field is selected, the message is encrypted when being posted to the URL specified.
Trust Store	If Is Secure is selected, enter the trust store for storing server side digital certificates. If you are using variables instead of the full path names ensure that the variable is defined in <code>yfs.properties</code> file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .

Table C–19 WebServices Configuration Properties

Property	Description
Trust Store Passwd	If Is Secure is selected, enter the password to access the trust store.
Key Store Type	If Is Secure is checked, set this value to JKS (Java Key Store).
Key Store	If Is Secure is selected, enter the key store for storing client side digital certificates. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Key Store Passwd	If Is Secure is selected, enter the password to access the key store.
SOAPActionURI	Enter the URI used by this attribute to invoke the required outbound web service. For example, <code>http://tempuri.org/PricingEngineGold/Service1/PricingEngineFunc</code> can be a valid input URI.
Arguments Tab	
Argument Name	The name of the parameter to be passed to the Webservice method.
Argument Value	The value of the parameter to be passed to the Webservice method.

Connection Properties

The following are the WebServices node’s connection properties:

Table C–20 WebServices Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except for FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any synchronous transport node Any other component node Any asynchronous transport node (except for FTP or File I/O); use a Pass-through node to connect them

Table C–20 WebServices Connection Properties

Connection	Node Connection Rules
Passes data unchanged	Transport nodes do not modify data

C.1.9 Synchronous WebLogic and MQSeries

The synchronous MQSeries and WebLogic transport nodes allow request and response operations using JMS queues. If the response is not received within a defined period, an exception is thrown.

Identification of appropriate response messages is done with a header field named MESSAGEID. When the request message is put into the queue, the MESSAGEID header is set to a unique value based on the current time and a counter. The response headers must have this same message ID for it to be picked up and processes correctly.

Note: In the case of MQSeries queues, when there are more threads running than messages available for pickup, the following message may appear in the adapter window:

```
2002.02.25 09:25:53 MQJMS2002E failed to get message
from MQ queue
```

Important: If you are running on IBM AIX and using MQSeries, include the following environment variable in the application server launch script, integration adapter script, and all agent server scripts:

```
LDR_CNTRL=MAXDATA=0x30000000
export LDR_CNTRL
```

Configuration Properties

The following are the properties of this node:

Table C–21 Synchronous WebLogic JMSQueue and MQSeries Properties

Property	Description
Runtime Tab	
Provider URL	<p>Enter the provider URL of the JMS implementation used. This is the URL to use for JNDI lookups.</p> <ul style="list-style-type: none"> For MQSeries using WebLogic, set the property (for file system context) to <code>file:[drive:]/<pathname></code> and ensure that the directory exists in your environment. For WebLogic JMS, set to <code>t3://<DNS Server Name or IP Address>:<port></code>.
Initial Context Factory	<p>The class name of the initial context factory. This is the starting point for the resolution of names for naming and directory operations.</p> <p>Select WebSphere MQ if you are using MQSeries accessed through a WebSphere iiop URL. This sets the class name to <code>com.ibm.websphere.naming.WsnInitialContextFactory</code>.</p> <p>Select File if you are using MQSeries accessed through a file URL, as with WebLogic. This sets the class name to <code>com.sun.jndi.fscontext.RefFSContextFactory</code>.</p> <p>Select WebLogic if you are using WebLogic JMS. This sets the class name to <code>weblogic.jndi.WLInitialContextFactory</code>.</p>
QCF Lookup	<p>Enter the queue connection factory name. This is used to retrieve the queue connection factory from JNDI. A client uses a queue connection factory to create queue connections with a JMS provider. Enter any unique identifier for the QCF lookup. This name must be the same as that configured in the WebLogic console or MQSeries.</p>
Needs Compression	<p>Select this option if the message needs to be compressed.</p>
Enable JMS Security	<p>Check this box if you want JMS Security to be enabled. Once selected, the JMS Security Properties tab is enabled to enter the name-value pairs.</p>
Response Tab	
Queue Name	<p>Enter the name of the queue in which the response message is received.</p>

Table C–21 Synchronous WebLogic JMSQueue and MQSeries Properties

Property	Description
Selector	Enter selectors based on the message headers. When specifying a selector, use only single quotes. For example, specifying the selector <code>APINAME='createOrder'</code> selects all messages with a header name= <code>'APINAME'</code> and value= <code>'createOrder'</code> .
Time Out (seconds)	Enter the time interval (in seconds) after which the requests time out.
Request Tab	
Queue Name	Enter the name of the queue to which the request message is sent.
Header Name	<p>The name of the message header. For example, <code>APINAME</code>.</p> <p>Choose  to add a new header name and value.</p> <p>Choose  to modify an existing header name and value.</p> <p>Choose  to delete an existing header name and value.</p>
Header Value	<p>The value associated with the Header Name. These name-value pairs are stored as message headers and can be queried by using message selectors.</p> <p>This can be set to a static value. For example, <code>'createOrder'</code> results in the message having a header <code>APINAME=createOrder'</code></p> <p>It can also be set to be dynamically extracted from the message using the syntax <code>xml://<full path of the element from root>/@<attribute name></code>, which results in the message with a header <code>APINAME='<value of attribute name in the XML>'</code>.</p>
Reconnect Tab	
Retry Interval (milliseconds)	In the event that the connection to the JMS server has been lost, enter the amount of time between each attempt to re-establish contact with the JMS server. This parameter is used in conjunction with the Number of Retries parameter. The default value is 0, implying no delay time between retry attempts.

Table C–21 Synchronous WebLogic JMSQueue and MQSeries Properties

Property	Description
Number of Retries	In the event that the connection to the JMS server has been lost, enter the number of attempts to re-establish contact with the JMS server before throwing an exception. This parameter is used in conjunction with the Retry Interval parameter. The default value is 0, implying no retries if the connection is lost and an exception is thrown immediately.
JMS Security Properties Tab	
This is enabled upon selecting Enable JMS Security in the runtime properties tab. You can override the JMS security properties specified here by enabling the agent and flow authorization parameters in <code>yfs.properties</code> .	
For more information on application server-specific security mechanisms see "Setting up the JMS Security Properties" section.	
Parameter Name	Enter the name of the security parameter.
Parameter Value	Enter the value of the security parameter.

Note: The JMS session objects can be pooled based on the service being executed. Hence, whenever the JMS sender requires a session object, the platform framework tries to get a free session object from the pool. If there are no free sessions available, a new session object is created to send the message and then added to the pool. Any session object that is idle for a certain configurable period of time is closed by the framework. For information on setting the properties for session reaptime check the `yfs.properties` file in the `<YFS_HOME>/resources` directory

Connection Properties

The following are the synchronous WebLogic JMSQueue and MQSeries nodes' connection properties:

Table C–22 Synchronous WebLogic JMSQueue and MQSeries Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> • Any transport node except for FTP or File I/O • Any other component node
Can be placed after	<ul style="list-style-type: none"> • Start node • Any synchronous transport node • Any other component node • Any asynchronous transport node (except for FTP or File I/O); use Pass-through node to connect.
Passes data unchanged	Transport nodes do not modify data

C.1.10 WebLogic and MQSeries JMS Queue

This component provides a common interface to invoke messaging services on WebLogic and MQSeries JMS queues. You can specify the required configuration properties in this component for enabling your JMS queues.

Note: In the case of MQSeries queues, when there are more threads running than messages available for pickup, the following message may appear in the adapter window:

```
2002.02.25 09:25:53 MQJMS2002E failed to get
message from MQ queue
```

Important: If you are running on IBM AIX and using MQSeries, include the following environment variable in the application server launch script, integration adapter script, and all agent server scripts:

```
LDR_CNTRL=MAXDATA=0x30000000
export LDR_CNTRL
```

C.1.10.1 WebLogic and MQSeries JMS Queue Sender

The JMS sender caches a single connection starting with the primary queue. When the message sending fails, the JMS sender retries to send the message a configurable number of times with a configurable delay between each attempt as defined in the Reconnect Tab of [Table C–23, "WebLogic and MQSeries JMS Sender Configuration Properties"](#). An exception is thrown if the retries are exhausted and there is no backup queue.

If a backup queue is setup, a connection is made to the backup queue. The backup queue becomes the current queue for the next message sending attempt. The sender can be configured to reconnect to the primary queue. If the primary queue is found to be working then a switch is made so that the primary queue becomes the current queue.

Configuration Properties

The following are the configuration properties of these nodes:

Table C–23 WebLogic and MQSeries JMS Sender Configuration Properties

Property	Description
Runtime Tab	
You need to use the settings in this tab for the primary JMS server.	
Queue Name	Enter the name of the queue to which the message is sent.

Table C–23 WebLogic and MQSeries JMS Sender Configuration Properties

Property	Description
Time To Live	<p>Enter the time period after which the messages are to be deleted from the queue. A value of zero causes the message to never be deleted from the queue.</p> <p>If this is set to a non-zero value, messages in the queue that are not consumed for the specified time interval are automatically deleted from the queue by the provider.</p>
Provider URL	<p>Enter the provider URL of the JMS implementation used. This is the URL to use for JNDI lookups.</p> <ul style="list-style-type: none"> • For MQSeries using WebLogic, set the property (for file system context) to <code>file:[drive:]/<pathname></code> and ensure that the directory exists in your environment. • For WebLogic JMS, set to <code>t3://<DNS Server Name or IP Address>:<port></code>
Initial Context Factory	<p>The class name of the initial context factory. This is the starting point for the resolution of names for naming and directory operations.</p> <p>Select WebSphere MQ if you are using MQSeries accessed through a WebSphere iiop URL. This sets the class name to <code>com.ibm.websphere.naming.WsnInitialContextFactory</code>.</p> <p>Select File if you are using MQSeries accessed through a file URL, as with WebLogic. This sets the class name to <code>com.sun.jndi.fscontext.RefFSContextFactory</code>.</p> <p>Select WebLogic if you are using WebLogic JMS. This sets the class name to <code>weblogic.jndi.WLInitialContextFactory</code>.</p>
QCF Lookup	<p>Enter the queue connection factory name. This is used to retrieve the queue connection factory from JNDI. A client uses a queue connection factory to create queue connections with a JMS provider. Enter any unique identifier for the QCF lookup. This name must be the same as that configured in the WebLogic console or MQSeries.</p>
Delivery Mode	<p>Indicate if the messages are to be Persistent or Non-Persistent when dropped into the queue.</p>
Needs compression	<p>Optional. Check this box if the message needs to be compressed before dropping into the queue.</p>

Table C–23 WebLogic and MQSeries JMS Sender Configuration Properties

Property	Description
Commit of this message depends on parent transaction	<p>Check this box if you want the message to be committed to the queue only after the service is completed.</p> <p>Uncheck this box if you want the message to be committed to the queue immediately.</p> <p>For example, if the ON_SUCCESS event of any standard Yantra 7x API is attached to a service in which the message is transactionally written to the queue, the message is committed to the queue only upon successful completion of the ON_SUCCESS event. The message is then rolled back from the queue if there is any error in the ON_SUCCESS event after the message is staged. However, in non-transactional mode, the message remains in the queue once it is staged and is not rolled back.</p>
Enable JMS Security	<p>Check this box if you want JMS Security to be enabled. Once selected, the JMS Security Properties tab is enabled to enter the name-value pairs.</p>
Header Tab	
Header Name	<p>The name of the message header. For example, APINAME.</p> <p>Choose  to add a new header name and value.</p> <p>Choose  to modify an existing header name and value.</p> <p>Choose  to delete an existing header name and value.</p>
Header Value	<p>The value associated with the Header Name. These name-value pairs are stored as message headers and can be queried by using message selectors.</p> <p>This can be set to a static value. For example, 'createOrder' results in the message having a header APINAME='createOrder'</p> <p>It can also be set to be dynamically extracted from the message using the syntax <code>xml://<full path of the element from root>/@<attribute name></code>, which results in the message with a header APINAME='<value of attribute name in the XML>'.</p>

Table C–23 WebLogic and MQSeries JMS Sender Configuration Properties

Property	Description
<p>Reconnect Tab</p> <p>You need to use the settings in this tab are for the backup JMS server.</p>	
<p>Retry Interval (milliseconds)</p>	<p>In the event that the connection to the primary JMS server is lost, enter the amount of time between attempts to re-establish contact with the primary JMS server. This parameter is used in conjunction with the Number of Retries parameter. The default value is 0 which means no delay time between retry attempts.</p>
<p>Number of Retries</p>	<p>In the event that the connection to the primary JMS server is lost, enter the number of attempts to re-establish contact with the primary JMS server before failing over to the backup JMS server, if enabled, or throwing an exception. This parameter is used in conjunction with the Retry Interval parameter. The default value is 0 which means there are no retries if the connection is lost and either failover occurs, if enabled, or an exception is thrown immediately.</p>
<p>Use backup JMS Queue</p>	<p>Check this box if you want to enable a backup JMS queue.</p> <p>Only upon selecting this checkbox, other controls in this tab are enabled.</p>
<p>Provider URL</p>	<p>Enter the backup JMS server provider URL of the JMS implementation used. This is the URL to use for JNDI lookups.</p> <ul style="list-style-type: none"> • For MQSeries using a file URL, set the property (for file system context) to <code>file:[drive:]/<pathname></code> and ensure that the directory exists in your environment. • For WebLogic JMS, set to <code>t3://<DNS Server Name or IP Address>:<port></code> • For WebSphere JMS, set to <code>corbaloc::<DNS Server Name or IP Address>:<bootstrap></code>

Table C–23 WebLogic and MQSeries JMS Sender Configuration Properties

Property	Description
Initial Context Factory	<p>The class name of the initial context factory for the backup JMS server. This is the starting point for the resolution of names for naming and directory operations.</p> <ul style="list-style-type: none"> • Select WebSphere MQ if you are using MQSeries accessed through a WebSphere iiop URL. This sets the class name to <code>com.ibm.websphere.naming.WsnInitialContextFactory</code>. • Select File if you are using MQSeries accessed through a file URL, as with WebLogic. This sets the class name to <code>com.sun.jndi.fscontext.RefFSContextFactory</code>. • Select WebLogic if you are using WebLogic JMS. This sets the class name to <code>weblogic.jndi.WLInitialContextFactory</code>.
Queue Name	Enter the name of the backup queue to which the message is sent.
QCF Lookup	Enter the back JMS server's queue connection factory name. This is used to retrieve the queue connection factory from JNDI. A client uses a queue connection factory to create queue connections with a JMS provider. Enter any unique identifier for the QCF lookup. This name must be the same as that configured in the WebLogic console or MQSeries.
Reconnect To Primary JMS Queue	Check this box if you want to reconnect to the primary JMS queue after a failover to the backup JMS queue server has occurred. Once this checkbox is selected, enter the wait time before it reconnects.
Time Before Reconnect (seconds)	Enter the maximum amount of time between attempts to re-establish contact with the primary JMS server after a failover to the backup JMS server. If contact is re-established with the primary JMS server, it will revert back to using the primary JMS server. If contact with the primary JMS server cannot be established, it will continue to use the backup JMS server. The default value is 600 seconds.

Table C–23 WebLogic and MQSeries JMS Sender Configuration Properties

Property	Description
JMS Security Properties Tab	
<p>This is enabled upon selecting Enable JMS Security in the runtime tab. You can override the JMS security properties specified here by enabling the agent and flow authorization parameters in <code>yfs.properties</code>.</p> <p>For more information on application server-specific security mechanisms see "Setting up the JMS Security Properties" section.</p>	
Parameter Name	Enter the name of the security parameter
Parameter Value	Enter the value of the security parameter.

Note: The JMS session objects can be pooled based on the service being executed. Hence, whenever the JMS sender requires a session object, the platform framework tries to get a free session object from the pool. If there are no free sessions available, a new session object is created to send the message and then added to the pool. Any session object that is idle for a certain configurable period of time is closed by the framework. For information on setting the properties for session reapttime check the `yfs.properties` file in the `<YFS_HOME>/resources` directory.

C.1.10.2 WebLogic and MQSeries JMS Receiver

The JMS receiver does not have a backup queue, therefore two different services must be configured. One to listen on the primary queue and one to listen on the backup queue. When an exception occurs, the receiver retries using an exponential back off mechanism with a configurable maximum wait limit between retries.

For example, if 600 seconds is the maximum wait time, it first waits for 1 second, then 2 seconds, 4, 8, 16 and so on exponentially until it reaches

600 seconds, after which it retries every 600 seconds. So there is no bound on the total wait time.

Since the JMS receivers operate independently of each other with respect to the primary and backup queues, the order of messages across these queues is not supported.

Configuration Properties

The following are the configuration properties of this node:

Table C–24 WebLogic and MQSeries JMS Receiver Configuration Properties

Property	Description
Runtime Tab	
Sub Service Name	Enter a unique identifier for each asynchronous receiver within a service definition.
Queue Name	Enter the name of the queue to which the message is sent.
Provider URL	<p>Enter the provider URL of the JMS implementation used. This is the URL to use for JNDI lookups.</p> <ul style="list-style-type: none"> For MQSeries using WebLogic, set the property (for file system context) to <code>file:[drive:]/<pathname></code> and ensure that the directory exists in your environment. For WebLogic JMS, set to <code>t3://<DNS Server Name or IP Address>:<port></code>
Initial Context Factory	<p>The class name of the initial context factory. This is the starting point for the resolution of names for naming and directory operations.</p> <p>Select WebSphere MQ if you are using MQSeries accessed through a WebSphere iiop URL. This sets the class name to <code>com.ibm.websphere.naming.WsnInitialContextFactory</code>.</p> <p>Select File if you are using MQSeries accessed through a file URL, as with WebLogic. This sets the class name to <code>com.sun.jndi.fscontext.RefFSContextFactory</code>.</p> <p>Select WebLogic if you are using WebLogic JMS. This sets the class name to <code>weblogic.jndi.WLInitialContextFactory</code>.</p>

Table C–24 WebLogic and MQSeries JMS Receiver Configuration Properties

Property	Description
QCF Lookup	Enter the queue connection factory name. This is used to retrieve the queue connection factory from JNDI. A client uses a queue connection factory to create queue connections with a JMS provider. Enter any unique identifier for the QCF lookup. This name must be the same as that configured in the WebLogic console or MQSeries.
Receiving Mode	<p>Indicate if the messages are received in a transactional mode or non-transactional mode.</p> <p>When this is set to non-transactional mode, the message is removed from the queue as soon as it is read.</p> <p>The messages are removed from the queue in a transactional mode only when the messages are functionally processed and if there is an exception while processing. The messages are then put into the YFS_REPROCESS_ERROR or YFS_INBOX tables.</p> <p>The messages are maintained in the queue if there is an exception when writing to the YFS_REPROCESS_ERROR or YFS_INBOX tables.</p>
Initial Threads	<p>Enter the number of threads that can process messages simultaneously. Based on your throughput, you can increase the number of threads to enhance performance using the System Management Console. For more information about using the System Management Console, see the <i>Yantra 7x System Management Guide</i>.</p> <p>You can also start multiple instances of the Yantra 7x Service Definition Framework for a specific integration adapter for a specific server. For more information, see the <i>Yantra 7x Performance Management Guide</i>.</p>
Selector	Enter selectors based on the message headers. These selectors must be in the form Header Name='Header Value'. When specifying a selector, use only single quotes. For example, using the selector APINAME='createOrder' selects all messages with a header name='APINAME' and value='createOrder'.

Table C–24 WebLogic and MQSeries JMS Receiver Configuration Properties

Property	Description
Service to Execute on EOF Message	Required if the message contains an EOF message ID. Choose  to select the service to be invoked when an EOF message is received. Once the EOF message is received, the framework waits for a few minutes (configurable in the <code>yfs.properties</code>) before executing this service. For more information see, Section C.1.5.3, "Enabling EOF Messages in the Platform Framework" .
Root Node Name of EOF Message	If the message contains an EOF message ID, enter your custom root node name for the EOF message. By default the end of file message has a root node of "EOF". For more information see, Section C.1.5.3, "Enabling EOF Messages in the Platform Framework" .
Enable JMS Security	Check this box if you want JMS Security to be enabled. Once selected, the JMS Security Properties tab is enabled to enter the name-value pairs.
Server Tab	
Server Name	Select the name of the server which actually executes the service. For more information on creating a new server refer to "Adding a New Server" .
Needs Decompression	Select this field to indicate that the incoming message is compressed and needs to be decompressed.
Exception Tabs	See the "Receiver Link Exception Handling Properties" table.
Reconnect Tab	
Maximum Time Between Retries (minutes)	Enter the maximum time between reconnection attempts. The default value is 10 minutes.
JMS Security Properties Tab	
This is enabled upon selecting Enable JMS Security in the runtime tab. You can override the JMS security properties specified here by enabling the agent and flow authorization parameters in <code>yfs.properties</code> . For more information on application server-specific security mechanisms see "Setting up the JMS Security Properties" section.	
Parameter Name	Enter the name of the security parameter.

Table C–24 WebLogic and MQSeries JMS Receiver Configuration Properties

Property	Description
Parameter Value	Enter the value of the security parameter.

Connection Properties

The following are the WLJMS and MQJMS nodes' connection properties:

Table C–25 WebLogic JMS and MQSeries JMS Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Yes, for services invoked both in a synchronous or asynchronous mode
Can be placed before	<ul style="list-style-type: none"> Any component node Any transport node (except for FTP or File I/O); use a Pass-through node to connect them
Can be placed after	<ul style="list-style-type: none"> Start node Any synchronous transport node Any other component node Any asynchronous transport node (except for FTP or File I/O); use a Pass-through node to connect them:
Passes data unchanged	Transport nodes do not modify data

C.1.11 Microsoft Message Queue (MSMQ)

The Microsoft Message Queue (MSMQ) transport node allows the Service Definition Framework to read or write messages to an MSMQ queue. MSMQ has the ability to use both local or remote queues. If a message is sent to a remote queue that cannot be found, an error is not generated. Instead, MSMQ creates a local queue of the same name. If the remote queue becomes available at a later time, the messages in the local queue are then transferred to the remote queue.

Note: MSMQ must be installed on the same system the integration adapter is running on.

C.1.11.1 MSMQ Sender

Configuration Properties

The following are the properties of this node:

Table C–26 MSMQ Sender Configuration Properties

Property	Description
Queue Name	Enter the name of the queue to which the message is to be sent using the following format: <machineName>/private\$/<QueueName> Yantra does not support Queue Lookup through the Active Directory.
Time To Live	Enter the time period after which the messages are to be deleted from the queue. A value of 0 causes the message to never be deleted from the queue. If this is set to a non-zero value, messages in the queue which are not consumed for the specified time interval are automatically deleted from the queue by the provider.
Persistent/Non Persistent	Select if the messages are to be Persistent or Non-Persistent when dropped into the queue.
Transactional/Non Transactional	Select Transactional if you want the message to be committed to the queue only after the service is completed. Select Non Transactional if you want the message to be committed to the queue immediately. For example, if the ON_SUCCESS event of any standard Yantra 7x API is attached to a service, in which the message is transactionally written to the queue, the message is committed to the queue only upon successful completion of the ON_SUCCESS event. The message is then rolled back from the queue if there is any error in the ON_SUCCESS event after the message is staged. However, in non-transactional mode, the message remains in the queue, once it is staged and is not rolled back. Note: If you plan to use transactional mode of messaging, create the queue in MSMQ as transactional queue.

Table C–26 MSMQ Sender Configuration Properties

Property	Description
String Message	Select this field if you want to set the PROPID_M_BODY_TYPE to VT_BSTR De-select this field if you want to set the PROPID_M_BODY_TYPE to VT_EMPTY.

Note: The JMS session objects can be pooled based on the service being executed. Hence, whenever the JMS sender requires a session object, the platform framework tries to get a free session object from the pool. If there are no free sessions available, a new session object is created to send the message and then added to the pool. Any session object that is idle for a certain configurable period of time is closed by the framework. For information on setting the properties for session reapttime check the `yfs.properties` file in `<YFS_HOME>/resources` directory.

C.1.11.2 MSMQ Receiver

Configuration Properties

The following are the properties of this node:

Table C–27 MSMQ Receiver Properties

Property	Description
Runtime Tab	
Runtime ID	Enter a unique identifier for each asynchronous receiver within a service definition.
Queue Name	Enter the name of the queue to which the message is to be sent using the following format: <machineName>/private\$/<QueueName> Yantra does not support Queue Lookup through the Active Directory.
Max Threads	The maximum number of threads for this process.

Table C–27 MSMQ Receiver Properties

Property	Description
Transactional/Non Transactional	Select if the messages are to be received in a Transactional or Non Transactional Mode. Note: If you plan to use transactional mode of messaging, create the queue in MSMQ as a transactional queue.
Server Tab	
Server Name	Select the name of the server which actually executes the service. For more information on creating a new server refer to "Adding a New Server" .
Exception Tabs	See Table C–29, "Receiver Link Exception Handling Properties" .

Connection Properties

The following are the MSMQ node’s connection properties:

Table C–28 MSMQ Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Yes, for services invoked both in a synchronous or asynchronous mode
Can be placed before	<ul style="list-style-type: none"> Any component node Any transport node (except for FTP or File I/O); use a Pass-through node to connect them
Can be placed after	<ul style="list-style-type: none"> Start node Any synchronous transport node Any other component node Any asynchronous transport node (except for FTP or File I/O); use a Pass-through node to connect them
Passes data unchanged	Transport nodes do not modify data

Note: Make sure the YFS_HOME/bin directory and the directory containing the jvm.dll are in your system PATH.

C.1.12 Receiver Link Exception Handling

For each asynchronous receiver nodes, enter exception parameters. Exception handling can be configured for the following nodes:

- [Database Receiver](#)
- [DCS 6.2 Database Receiver](#)
- [File I/O Receiver](#)
- [FTP Receiver](#)
- [WebLogic and MQSeries JMS Receiver](#)
- [MSMQ Receiver](#)

Configuration Properties

The following are the exception properties of the asynchronous receiver nodes:

Table C–29 Receiver Link Exception Handling Properties

Property	Description
Alert Type	Enter the type of Alert being raised when an exception occurs. For example, you can enter the text OrderCreate. This appears in the Alert console and can be used to filter particular type of alerts.
Alert Queue Name	Select the name of the alert queue to which the exceptions are sent.
Suspend API	Select this field if a suspendable exception is returned by an extended API, the message is retained in the queue and the execution restarts after the Suspend Wait Time interval. For details regarding the exception to be thrown, see the YIFRestartableAPI interface in the <i>Yantra 7x Javadocs</i> .
Suspend Wait Time	Enter the time to wait before attempting to reprocess the message.

Table C–29 Receiver Link Exception Handling Properties

Property	Description
Is Reprocessible	Select this field if the message received from an asynchronous source (like a message queue or database) and the error XML must be saved in the exception console when an exception occurs. Messages marked as Reprocessible can be corrected in the Exception Console and submitted for reprocessing. For more information about the Exception Console, see the <i>Yantra 7x Platform User Guide</i> .
Check for Prior Exception	Select this field to check for prior exceptions before the execution of the service. Choosing this option implies that prior to executing the service, a check is made to see if any related errors exist for the message. This check must be implemented externally through the <code>YIFErrorSequenceUE</code> user exit. Note: This option is applicable only when all the related services are associated to the same server.
Exception Group	Enter a group of related services where exceptions are linked. For example, two services for receiving modifications on an order from external systems. Note: This option is applicable only when all the related services are associated to the same server.
Prior Errors User Exit	Enter the class name that implements <code>YIFErrorSequenceUE</code> user exit for checking prior related errors for the message. For more information on <code>YIFErrorSequenceUE</code> refer to the <i>Yantra 7x Javadocs</i> .
Exception References Tab	
Exception Reference Name	Enter the name of the exception reference. Saved in the <code>ERROR_REFERENCE</code> column of the <code>YFS_REPROCESS_ERROR</code> table to indicate as <code>Name=Value</code> . For example, <code>NAME1</code> .

Table C–29 Receiver Link Exception Handling Properties

Property	Description
Exception Reference Value	<p>Enter the associated Exception Reference value. These name value/pairs are stored in the ERROR_REFERENCE field in the YFS_REPROCES_ERROR table for querying purposes.</p> <p>Can be set to be static. For example, entering '1234' results in the ERROR_REFERENCE field to be populated with NAME1=1234.</p> <p>Can also be set for dynamically extraction from the message using the following syntax <code>xml://<full path of the element from root>/@<attribute name></code>.</p> <p>For example, to get the sales order number from the createOrder input XML, use <code>xml://Order/@OrderNo</code>. This results in the ERROR_REFERENCE field to be populated with NAME1=<value of attribute OrderNo in the XML>.</p>

C.2 Component Nodes

Component nodes determine how data should be transformed from one format to another. To define their configuration properties, by clicking the node and editing the properties as described below.

For detailed information about each component node, see the following topics:

- [Alert](#)
- [Application Programming Interface \(API\)](#)
- [Composite Service](#)
- [Condition](#)
- [E-Mail](#)
- [Nomenclature Runtime Component](#)
- [Router](#)
- [Text Translator](#)
- [XSL Translator](#)
- [Defaulting Component](#)
- [Data Security](#)

- [Jasper Printer Component](#)

C.2.1 Alert

This component allows Alerts to be registered in the Alert console. This is same as invoking alerts from an Action in releases prior to 5.0. You can also consolidate the alerts by grouping the attributes in the YFS_INBOX table.

Configuration Properties

The following are the properties of this node:

Table C–30 Alert Configuration Properties

Property	Description
General Tab	
Alert Queue Name	Select the name of the queue alerts should be sent to.
User ID	Select the user ID to which the alert is assigned.
Alert Type	Enter the type of the alert. The type that is assigned here appears as the Alert Type in the Alert Console and can be used to filter alerts. The type entered is user defined.
Priority	Enter the priority of the exception raised. Defaults to 0.
Description	Enter a brief description of the alert raised.
Template Name	Enter the XSL Template Name. The output of the XSL and the incoming XML document merge is displayed in the description field on the Alert Console.
List Template	Enter the XSL list template. The output of the XSL and the incoming XML document merge is displayed on the Alert List and Home Page.
Resolve By (Hrs)	Enter the number of hours the alert should be resolved by. The hours can be specified in decimals also, for example 5.5 or 20.
Expiration Days	Specify the number of expiration days for the alert after which this exception may be automatically closed. A value of zero means the exception does not expire.

Table C–30 Alert Configuration Properties

Property	Description
System Arguments Tab	
Argument Name	Predefined elements available to a particular process repository. The following attributes are optional and can be specified as static values or dynamic values. xml://<ElementName>/@<AttributeName> InboxType: ItemId: EnterpriseKey: OrderHeaderKey: OrderLineKey: OrderNo: ShipnodeKey: SupplierKey: LoadNo: ShipmentNo
Argument Value	The argument value.
User Arguments Tab	
Reference Name	Enter additional name/value information that can be used to query the exceptions.
Reference Value	Enter the reference value.
Reference Type	Enter the reference type. Valid values are: TEXT, URL
Consolidation Tab	
Consolidation Required	Check this box if you wish to consolidate the alerts. A consolidation count is kept to increment similar alerts that follow the same template attributes. If it is not selected, then the count always remains 1.

Table C–30 Alert Configuration Properties

Property	Description
Consolidation Template	<p>Select either Default or Custom for consolidating the alerts. The default chosen option is Default.</p> <p>If Custom is chosen, then the attributes for the consolidation template appear below. These attributes are from the YFS_INBOX table and the corresponding XML file can be found in <YFS_HOME>/template/api/exceptionConsolidation.xml. You can select any of the attributes by right clicking the attribute and including it for consolidation.</p> <p>If you have extended the columns in YFS_INBOX table, then those attributes appear under the Extn element.</p>
Consolidate Dates By	<p>Specify whether the consolidation is done daily or hourly. By default the option is set to Day.</p> <p>Choosing this option is useful only if you include any time attributes such as Date in the consolidation template.</p>

Connection Properties

The following are the alert node's connection properties:

Table C–31 Alert Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> • Any transport node except for FTP or File I/O • Any other component node
Can be placed after	<ul style="list-style-type: none"> • Start node • Any transport node except for FTP or File I/O • Any other component node
Passes data unchanged	Yes

C.2.2 Application Programming Interface (API)

The API component is used to invoke the Yantra 7x System APIs or any user-written java class.

Note: To configure extended database APIs for custom and hang-off entities refer to *Yantra 7x Customization Guide*.

Configuration Properties

The following are the properties of this node:

Table C–32 API Configuration Properties

Property	Description
General Tab	
Standard Yantra API	Select this option if a standard Yantra 7x API is to be invoked. If selected, a Standard Yantra 7x API Name drop down list is displayed. For each API, the Class Name and Method Name are provided and cannot be edited.
Extended API	Select this option if a custom java code is to be invoked.
Extended Database API	Select this option if the service invokes a custom or hang-off API. If selected, a custom API Name drop-down list is displayed. For each API, the Class Name and Method Name are provided and cannot be edited.
API Name	Select or enter the API to be called. Note: This field is for integration purposes only.
Class Name	Specifies the class to be called.
Method Name	Specifies the method to be called.
Requires Backward Compatibility	Select this field to indicate that input data coming through the API is from a previous version (only applicable to Yantra 7x system APIs).
Version	If you chose Requires Backward Compatibility, select the Yantra version the API is to behave as. Only the applicable versions for the individual API are displayed.
Arguments Tab	

Table C–32 API Configuration Properties

Property	Description
Argument Name	You can pass name/value pairs to the API by entering the values in the Arguments Tab. In order for custom APIs to access custom values, the API should implement the interface <code>com.yantra.interop.japi.YIFCustomApi</code> . If entered, these name/value pairs are passed to the CustomApi as a Properties object.
Argument Value	Enter the argument value.
Template Tab	
	When Yantra 7x System APIs are invoked, you can specify an output template to be used by the API. You can specify the template in the configuration properties of the Service Definition, the Resource Definition in the Resource Configurator, or both. However, if the template has been specified at both definition levels, the template specified in the Service Definition is used.
XML Template	Select this radio button to construct the XML to be used for the API output. Enter the template root element name and click OK. You can then construct the XML.
File Name	Select this radio button to enter the filename of the XML file to be used as the API output template. This file should also exist in your CLASSPATH.

Connection Properties

The following are the API node's connection properties:

Table C–33 API Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any transport node except FTP or File I/O Any other component node

Table C–33 API Connection Properties

Connection	Node Connection Rules
Passes data unchanged	Yes

C.2.3 Composite Service

The Composite service node enables users to specify multiple services that need to be executed as part of a single service. Only services that can be invoked synchronously can be part of the Composite service.

Note: If a transaction fails after executing a service in a composite component which drops the message into WebLogic JMS, MQ JMS, DB or FILE I/O. The message in the asynchronous medium cannot be rolled back.

Caution: Never configure a composite service to call its parent service if not it leads to an infinite loop.

Table C–34 Composite Service Configuration Properties

Property	Description
Service Name	Select services to be invoked by this service.

After you add a service, you can view and edit its properties. On the list of services, double-click the name of the Composite service you want to view or edit. It opens in a new window.

Connection Properties

The following are the Composite Service node's connection properties:

Table C–35 Composite Service Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except for FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any transport node except for FTP or File I/O Any other component node
Passes data unchanged	Yes

C.2.4 Condition

This node is invoked synchronously. Conditions allow you to build branching logic within the service.

Note: Do not place a Condition node very close to other objects (nodes or connectors), otherwise the outbound connector points may not become active.

Configuration Properties

The following are the properties of this node:

Table C–36 Condition Configuration Properties

Property	Description
Condition Name	Select a condition from a list of pre-configured condition names or you can create a new Condition.

Connection Properties

The Condition node requires two outgoing connections, one for true circumstances and one to false. When you create the outgoing links, *first* link to the node that processes the true condition and *then* link to the node that processes the false condition.

The following are the Condition node's connection properties:

Table C–37 Condition Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except for FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any transport node except for FTP or File I/O Any other component node
Passes data unchanged	Yes

C.2.5 E-Mail

The E-Mail component type wraps the SMTP protocol. It can be invoked synchronously or asynchronously. The configuration allows you to specify either the addresses (From, To, and so forth) from the XML should be dynamically retrieved or sent to a static address.

Note: If the e-mail being sent contains HTML content, the XSL that is transforming the input XML to the HTML format should have a comment such as the following example:

```
<xsl:comment>CONTENT_TYPE=text/html</xsl:comment>, so the transformed HTML has a comment: <!--CONTENT_TYPE=text/html-->
```

The framework uses this information to set the content_type to text/html, if this comment is not present the content_type is set to text/plain.

Configuration Properties

The following are the properties of this node:

Table C–38 E-Mail Configuration Properties

Property	Description
E-mail Server	Enter the name or IP address of the mail server.
E-mail Server Listener Port	Enter the port number of the mail server.

Table C–38 E-Mail Configuration Properties

Property	Description
Subject	Enter what you want to appear in the subject line of the e-mail. If you specify XML in the format of <code>xml://<ElementName>/@<AttributeName></code> , it is replaced dynamically with the value from the input XML data. For example, the text “Thank you for your online Order <code>xml://Order/@OrderNo</code> ” is a combination of static and dynamic text that results as “Thank you for your online Order MyOrder005”.
Body Template	XSL file that contains formatting to apply to the body of the message. Yantra 7x supplies the <code>template/email/orders_mail.xsl</code> file.
From	Can be static or dynamic with the XML path specified as <code>xml://<ElementName>/@<AttributeName></code> . Use semi-colons as a delimiter between addresses.
To	Required. Can be static or dynamic with the XML path specified as <code>xml://<ElementName>/@<AttributeName></code> . Use semi-colons as a delimiter between addresses.
CC	Can be static or dynamic with the XML path specified as <code>xml://<ElementName>/@<AttributeName></code> . Use semi-colons as a delimiter between addresses.
BCC	Can be static or dynamic with the XML path specified as <code>xml://<ElementName>/@<AttributeName></code> . Use semi-colons as a delimiter between addresses.

Connection Properties

These are the connection properties of this node:

Table C–39 E-Mail Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except for FTP or File I/O Any other component node

Table C–39 E-Mail Connection Properties

Connection	Node Connection Rules
Can be placed after	<ul style="list-style-type: none"> • Start node • Any transport node except for FTP or File I/O • Any other component node
Passes data unchanged	Yes

Note: The SMTP connection objects can be pooled based on the service being executed. Hence whenever the SMTP sender require a connection object the platform framework tries to get a free connection object from the pool. If the connection objects in the pool are occupied or if the pool is empty, then a new connection object is created to send the message and then added to the pool. Any connection object idle for a certain configurable period of time and can be closed by the framework. For information on setting the properties for connection reptime check the `yfs.properties` file in `<YFS_HOME>/resources` directory.

C.2.6 Nomenclature Runtime Component

The Nomenclature Runtime component provides a mapping tool that allows you to configure unique terms you use to match unique terms your trading partners use.

Configuration Properties

The following are the properties of this node:

Table C–40 Nomenclature Runtime Properties

Property	Description
Nomenclature XML Name	Defines the source to destination mapping for documents needing transformation.

C.2.6.1 Nomenclature Runtime

For configuring the runtime properties:

Table C–41 Nomenclature Runtime Configuration Properties

Property	Description
XML Name	Select XML name created through nomenclature configuration. For more information about nomenclature configuration, see the <i>Yantra 7x Platform Configuration Guide</i> .

Connection Properties

The following are the Nomenclature Runtime connection properties of this node:

Table C–42 Nomenclature Runtime Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except for FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any transport node except for FTP or File I/O Any other component node
Passes data unchanged	No

C.2.7 Router

Allows business documents to be routed based on participant preferences. Participants can be configured with different services to enable business documents to be delivered to them. In the scenario modeling, data published to trading partners is marked through a Router.

Router extracts the organization code from the data published during run time and extracts relevant organization preferences for document delivery and executes the service specified for the Participant.

Configuration Properties

The following are the properties of this node:

Table C–43 Router Configuration Properties

Property	Description
Document Name	Select the document being routed to the Participant. For example, Purchase Order.
Route XML Data To Following Roles	
Trading Partner Role	Select the participant's role. For example, BUYER or SELLER
XML Attribute	Enter the XML path where the participant's organization code is located. For example, xml://Order/@OrganizationCode. Participant preferences are appended to the document before executing the specified service.

Connection Properties

The following are the Router node's connection properties:

Table C–44 Router Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any transport node except FTP or File I/O Any other component node
Passes data unchanged	Yes

C.2.8 Text Translator

The Text Translator converts flat files to and from XML format, enabling flat-file applications to integrate with Yantra 7x. For detailed information about Text Translator file configuration, see [Appendix D, "Text Translator Reference"](#).

Configuration Properties

The following are the properties of this node:

Table C–45 Text Translator Configuration Properties

Property	Description
Input Data Format	Select the format in which data is input. Options are: <ul style="list-style-type: none"> Text - Flat text files XML - Structured text files
Output Data Format	Enter the format of the output data, automatically determined by the choices for Input Data Format combined with Input Text Format or Output Text Format.
Input Text Format	If you selected Input Data Format to be Text select: <ul style="list-style-type: none"> Text Positional - For a Flat text file with fields that have a fixed maximum length and records that have a common end-of-record terminator Text Delimited - For a Flat text file that contains one or more records separated by a specified delimiter, or separator
Schema Name	Enter the relative path to the schema description file of the translation from XML input to a flat file. The location is relative to the CLASSPATH of the integration adapter. For example, if the flat file schema XML file is located in <YFS_HOME>/bin/test.xml, and <YFS_HOME> is in the CLASSPATH, then the value of the attribute is /bin/test.xml.

Connection Properties

A Text Translator has to be either preceded by a File or DCS 6.2 Database node or succeeded by a File or DCS 6.2 Database node, using the following conditions:

- When the File or DCS 6.2 Database node is *followed* by a Text Translator component:
 - Output format is always XML
 - Input format can be text or XML
- When the Text Translator component is *followed* by a File or DCS 6.2 Database node:
 - Output format must be text (either positional or delimited)
 - Input format must be XML

The Text Translator component cannot be directly before or after a pass-through node.

The following are the Text Translator node's connection properties:

Table C-46 Text Translator Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any synchronous transport node Any other component node except for a pass-through node
Can be placed after	<ul style="list-style-type: none"> Start node Any synchronous transport node Any other component node except for a pass-through node
Passes data unchanged	Yes

C.2.9 XSL Translator

The Extensible Stylesheet Language (XSL) is used to transform XML documents into display formats such as HTML.

In Yantra 7x, a classpath can be used to find the XSL files included by the `xsl:include` directive. To enable this, a new property `yfs.xsl.uriresolver` is added to the `yfs.properties` file located in `<YFS_HOME>/resources/` directory. For more information on the proper usage of this property see the `yfs.properties` file.

Configuration Properties

The following are the properties of this node:

Table C-47 XSL Translator Configuration Properties

Property	Description
XSL Name	Enter the XSL template name. The location is relative to the CLASSPATH of the integration adapter.

Connection Properties

The following are the XSL Translator node's connection properties:

Table C–48 XSL Translator Connection Properties

Connection	Node Connection Rules
Can be the first node after the start node	Only for services invoked synchronously
Can be placed before	<ul style="list-style-type: none"> Any transport node except for FTP or File I/O Any other component node
Can be placed after	<ul style="list-style-type: none"> Start node Any transport node except for FTP or File I/O Any other component node
Passes data unchanged	Yes. The XSL translator can manipulate data streams to fit specific business integration needs.

C.2.10 Defaulting Component

This component applies defaulting based on configured properties and invokes a class to apply additional overrides. It can also be used to localize data in the XML.

The input to this component is an XML and the output is the same XML with the defaults applied. The properties of this component define the XML attributes to which the defaults are applied. The attributes themselves are defined through the notation used in other service components.

For example, in the `createOrder()` API assume the unit of measure on the `<Item>` element in the `<OrderLine>` node needs to be defaulted. The component property is defined as:

```
Attribute=/Order/OrderLines/OrderLine/Item/@UnitOfMeasure
Default Value = "EACH"
Override = Y
```

If the input XML is:

```
<Order>
  <OrderLines>
    <OrderLine>
      <Item ItemID="" />
    </OrderLine>
```

```

    </OrderLines>
</Order>

```

A UnitOfMeasure attribute is added with a value of EACH to the Item element. The override property indicates if the attribute in the input XML needs to be overridden irrespective of the existence of the attribute.

The configuration properties for the defaulting component defined in [Table C–49](#) are optional.

Table C–49 Defaulting Component Configuration Properties

Property	Description
General Tab	
Defaulting Template	<p>Enter the path for the defaulting template. This template consists of a set of attributes and their default values. For example:</p> <pre> <Overrides> <Override AttributeName="/Order/OrderLines/OrderLine/Item/@UnitOfMeasure" AttributeValue="EACH" Override="Y" /> </Overrides> </pre> <p>Note that the attribute path is an XPath variable.</p>
Custom Class	<p>Enter the custom class that provides the defaulting attributes. This class gets the original input to the component, modified XML by the defaulting template and apply any additional overrides and return the modified output XML. This class implements the <code>YIFXMLAttrOverride</code> interface. For more information on this interface refer to the <i>Yantra 7x Javadocs</i>.</p> <p>This class can also verify the existence of order and apply or override the defaults applied previously. This class is primarily used to default based on lookup of data from the database or to reverse the defaulting, in the case of a modification.</p>

Table C–49 Defaulting Component Configuration Properties

Property	Description
LocaleCode Path	<p>Enter the locale code path within the input document as an XPath expression.</p> <p>You can specify the Localize Attribute in the custom overrides rather than specifying a static attribute value. For example the XML file can contain the type as:</p> <pre><Overrides> <Override AttributeName="//Organization/@OrganizationCode" Type="LOCALIZE" /> </Overrides></pre>
Custom Overrides Tab	
Element Path	Specifies the path of the element for the custom overrides. This is an Xpath expression.
Attribute	Specifies the name of the attribute.
Value	Specifies the value of the attribute.
Adding a Custom Override	
Element Path	Enter the Xpath of the element for the custom overrides.
Attribute Name	Enter the name of the XML attribute that you wish to override.
Localize Attribute	Select this option if the attribute value is localized. The path for localization is specified in the LocaleCode.
Use Static Attribute Value	Select this option if you want to use a static attribute value. Once this option is selected you can enter the value and check the override for an existing value.
Attribute Value	Enter the value for the attribute.
Always override Existing value	Check this box if you want to always override the existing value with the new value entered above.

C.2.11 Data Security

The data security component enables users to configure an attribute of the XML that is coming in or flowing through the service to be validated against the list of enterprises. This enables you to secure data based on

the data security groups discussed in [Section 5.3, "Defining Data Security Groups"](#).

For enterprise validation, the list of enterprises that the user can access is determined and validated for the attribute value provided. Once the user is determined, the data security ID for that user is picked from the `YFS_USER` table and the list of enterprises the user can access is determined and the value of the attribute is validated.

For custom validation, you can implement a `YIFSecurityValidator` interface to set and validate the user ID. The class that implements this interface is created, the user is set in that component, and the input XML is parsed to obtain a list of attribute values, which are then passed to a validating method.

For more information on the `YIFSecurityValidator` see the *Yantra 7x Javadocs*.

If the security component access validation succeeds, the input XML is passed to the next component. If the validation fails an error is thrown back to the caller indicating that the security access failed.

This data security component can be used in a service where an XML is flowing through. If the component is configured after a component that does not output an XML, a runtime error is thrown.

Table C–50 Data Security Panel Configuration Properties

Property	Description
General Tab	
Enterprise Validation	Select this option if you want the user group to be validated against a list of enterprises. If this option is selected, the Attributes to Validate must be entered.
Attributes to Validate	Required, only if enterprise validation is selected. Enter the XPath to an attribute that needs to be validated.
Custom Validation	Select this option if you choose to do a custom validation. The class name should be provided if this option is selected.
Class Name	Required, only if custom validation is selected. Enter the name of the custom class that you wish to implement for the custom validation.

Table C–50 Data Security Panel Configuration Properties

Property	Description				
User Identification	Select one of the following options to set the user information for the custom or enterprise validation. <table border="1" data-bbox="672 366 1269 534"> <tr> <td>User ID</td> <td>Select the user ID from the drop-down list.</td> </tr> <tr> <td>YFSEnvironment</td> <td>Select this option to use the user ID from YFSEnvironment.</td> </tr> </table>	User ID	Select the user ID from the drop-down list.	YFSEnvironment	Select this option to use the user ID from YFSEnvironment.
User ID	Select the user ID from the drop-down list.				
YFSEnvironment	Select this option to use the user ID from YFSEnvironment.				
Arguments Tab					
Argument Name	The name of the parameter to be passed to the validator method.				
Argument Value	The value of the parameter to be passed to the validator method.				

C.2.12 Jasper Printer Component

This component is used to automatically print a document based on an event. It is a standard XML-based component: accepts XML as input and provides an identical output XML.

For example, printing a pick list in a store for store pick-up is an example where the store does not maintain the inventory. In this case, the website where the order is placed or a call center captures orders for store pick-up. These orders are sent down to the store for processing. A store inventory control associate periodically checks to see if there are any orders to be picked. If there are, the associate can print a paper pick list to pull the products required off the retail floor. In some cases, the print can be triggered automatically when an order is received at the store.

A document can also be printed conveniently by a single click from the console. This is supported by using flow execution and the printer component.

You can also audit the success and failure of the printing events using the print transaction defined under the general process type. This transaction is configurable and has two events: Print.ON_SUCCESS and Print.ON_FAILURE. The ON_FAILURE event is raised only for service suspend exceptions such as print failures.

The following table provides the Jasper printing configuration properties:

Table C-51 Jasper Printing Configuration Properties

Field	Description
Jasper Report	This is a compiled Jasper file. The file name supports variables in both a standard and enhanced variable format as explained in <i>Yantra 7x Customization Guide</i> .
Report Select Exp	Specify the element to be sent as the root of the report. By default, the entire XML is used. An error is thrown, if this field is given but does not resolve to an element.
Printer Name	Enter the XML path pointing to the printer to use.
Locale	Specify the locale for translating the literals in the report. The data in the XML can be localized using the defaulting component discussed in Section C.2.10, "Defaulting Component" .
Variables Tab	
Variable Name	The name of the parameter to be passed to the validator method.
Variable Value	The value of the parameter to be passed to the validator method.

To allow custom print formats, the configuration of the print component supports changing report file names based on an input XML. For example, if the input XML contains the file name:

```
${jasperfolder}/compiled/${orgcode}/report.${doctype}.jasper
```

First, all variables are resolved against the variables defined in the Variables tab:

```
orgcode → xml:/Order/@OrganizationCode
```

```
doctype → xml:/Order/@DocumentType
```

The remaining variables are resolved against the `yfs.properties` file. In this case, the `${jasperfolder}` variable is resolved from the `yfs.properties` definition: `jasperFolder=/someCustomJasperFolder`.

So, with the input of `<Order OrganizationCode="DEFAULT" DocumentType="0001" />` the expression resolves to: `/someCustomJasperFolder/compiled/DEFAULT/report.0001.jasper`.

An exception is thrown if an error occurs when processing the report. For example, an exception is thrown if the file name is invalid. A service suspend exception is thrown while printing the report enabling the ability to pause an asynchronous service.

C.3 Adapter Nodes

The Yantra 7x Service Definition Framework provides hooks to connect to external systems like Sterling GIS through custom adapters. Yantra 7x currently provides the following standard adapter:

- [Sterling GIS](#)

C.3.1 Sterling GIS

This adapter is used for communicating with the Sterling GIS product over the HTTP protocol. It is capable of streaming XML over HTTP connection as opposed to sending name-value pairs resulting in better performance. However, if you want to use name-value pairs you need to be on a specific GIS fix pack. For more information on the fix pack appropriate for your JDK version see *Yantra 7x Installation Guide*.

Configuration Properties

The following are the properties of this adapter:

Table C–52 Sterling GIS Configuration Properties

Property	Description
URL	Enter the URL to which the message is to be posted.
Stream XML Document	Select this field if you want to send and receive streaming XML documents. If selected, the "HTTP Post Variable" entry box is disabled.
HTTP Post Variable	Enter the variable name to which the HTTP post data is to be assigned.
Is Secure	If this field is selected, the message is encrypted when being posted to the URL specified.
Key Store Type	If Is Secure is checked, set this value to JKS (Java Key Store).

Table C–52 Sterling GIS Configuration Properties

Property	Description
Key Store	If Is Secure is selected, enter the key store for storing client side digital certificates. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Key Store Passwd	If Is Secure is selected, enter the password to access the key store.
Trust Store	If Is Secure is selected, enter the trust store for storing server side digital certificates. If you are using variables instead of the full path names ensure that the variable is defined in yfs.properties file. For more information on including these variable see <i>Yantra 7x Customization Guide</i> .
Trust Store Passwd	If Is Secure is selected, enter the password to access the trust store.

If you want to use name-value pairs enter the HTTP post variable name. If you need to specify any additional name-value pairs enter the name and value in the Post Variables tab. In this tab, select  and follow the information given in the [Table C–53](#):

Table C–53 Sterling GIS Post Variable Properties

Fields	
Post Variable Name	Enter the name of the post variable.
Post Variable Value	Enter the value of the post variable name.

C.4 Installing the Service Builder Samples

Yantra 7x provides several samples of services that can be configured for different business scenarios.

To install the service builder samples:

1. If the `<YFS_HOME>/resources` directory does not have a `migrator.properties` file, then copy the `<YFS_HOME>/resources/migrator.properties.sample` file to `<YFS_HOME>/resources/migrator.properties`.

2. Set the `yfs.migrator.xml.directory` property to `<YFS_HOME>/database/FactorySetup/ServiceSamples`.
3. Run `<YFS_HOME>/bin/migrator.sh <YFS_HOME>/resources/servicesamples.xml`.

Text Translator Reference

Occasionally, service nodes require files to be configured, in addition to the graphical user interface. This appendix describes the files you need to configure for the Text Translator node.

D.1 Understanding File Formats

A flat file usually contains a series of records (or lines), where each record is usually a sequence of fields.

A flat file format can be any of the following types:

- Positional
- Delimited
- XML

Positional Flat Files

A positional flat file has fields that have a fixed maximum length and records that have a common end-of-record terminator.

For example, if a positional flat file on a UNIX system has a Name field that is fixed at a maximum of fifteen characters, a Color field that is fixed at a maximum of ten characters, white space padding the end of each field, and the `\n` new line feed character, it would appear as shown in the example below:

Example D–1 Fragment of a Positional Flat File

```
NAME           COLOR      \n
Sam            turquoise \n
David          red        \n
```

```
Elizabeth      orange      \n
```

Delimited Flat Files

A delimited flat file contains one or more records set off from each other by a specified delimiter, or separator. For example, each record may be terminated by the operating system's line separator and each field within a record may be separated by a comma.

Delimiters are not read in as part of the data. However, if the delimiter character does appear as data, the data can be formatted so the data and the delimiter are distinguishable. For example, the field in which a delimiter character appears can be enclosed in quotation marks to indicate that the delimiter character is to be treated as data and not as a delimiter. For example, if you choose to use an asterisk (*) as the delimiter and it also appears in a data field following the word Special, then it should exist in the flat file as "Special*". In other words, the complete data field should be within quotation marks (" ").

For example, if a delimited flat file on a UNIX system has fields that are delimited by commas, it would appear as shown below.

Example D–2 Fragment of a Delimited Flat File

```
NAME, COLOR  
Sam, turquoise \n  
David, red \n  
Elizabeth, orange \n
```

XML Flat Files

In some cases, legacy applications produce XML files. These native XML files need to be translated into a format that is usable by Yantra 7x by means of an XSL.

If the input flat files are XML files, then the source XML must contain a root element, followed by a sequence of one or more child elements. Each child element under the root element is treated as the input for the flat file receiver service. For example, if the source XML is like so:

```
<?xml version="1.0" encoding="UTF-8"?>  
<Orders>  
  <Order name="A1">...</Order>  
  <Order name="A2">...</Order>  
  <Order name="A3">...</Order>
```

```
</Orders>.
```

then the input XML is processed to create three different XMLs, as follows:

```
<Order name="A1" />
<Order name="A2" />
<Order name="A3" />
```

The API that has been configured for this flat file receiver is executed three times, each with a different XML as shown above.

For examples of flat files using the `createOrder()` API as input, see the `<YFS_HOME>/documentation/code_examples/flatfile/` directory.

The location of the data in the flat file is specified as the incoming directory within the configuration. The flat file is used as input to a Yantra 7x API, as determined by the XSD.

D.2 Text Translator Components

The Text Translator uses the following types of files:

- [XSD Files](#)
- [Schema Files](#)

D.2.1 XSD Files

The Text Translator uses XSD files, a type of XML file, to convert flat files to and from XML format. Yantra 7x supplies four XSDs—two XSDs for data coming in from positional and delimited files and two XSDs for data from Yantra 7x to convert to positional and delimited flat files.

These files follow the standards defined by the World Wide Web Consortium (W3C); for information on this standard see <http://www.w3.org/XML/Schema>.

All four XSD files are designed specifically to correspond with Yantra 7x APIs, and so they cannot be modified. However, you do need to refer to the XSDs when defining and implementing schema files.

Each of the following XSD files is described within this chapter:

- [Text Translator Positional Receiver XSD File](#)

- [Text Translator Delimited Receiver XSD File](#)
- [Text Translator Positional Sender XSD File](#)
- [Text Translator Delimited Sender XSD File](#)

These files are archived in the `ycpbe.jar` file in order to be used by Yantra 7x.

D.2.2 Schema Files

The Text Translator uses schema files, a type of XML file, that describe the structure of each flat file. The format of the schema determines how the data is translated by a service and should unambiguously determine the parsing of the flat file data.

File schemas are required on a per API per file format basis (for example, you may need to create a positional `createOrder` schema or a delimited `createOrder` schema), so you need to write least one unique schema per API you intend to execute.

The translation step can be independent of the source of the input data. The same data could arrive through a flat file store or through a JMS queue, and the translation specification applies in either case.

The schema file for byte count check is 0-based for the file header and 1-based for the file trailer. Therefore, you must take this into account when creating the file schema. When specifying the `FileTrailerLength` in the positional schema file, you must account for a new line if the file trailer ends with a new line. On UNIX, a new line has one byte, and on Windows a new line has two bytes. The length of the new line character has to be added to the `FileTrailerLength`.

The schema file is stored in any location accessible as a `CLASSPATH` variable. For sample schema files that correspond to the provided sample input files, see the files in the `<YFS_HOME>/documentation/code_examples/flatfile/` directory.

Note: Yantra 7x recommends that you periodical check for files within the `/error/` and `/complete/` directories that contain errors or remain unprocessed.

D.2.3 Defining a Schema File

The schema file describes the structure of your incoming data file. The format of the schema determines how the data is translated by the service and should unambiguously determine the parsing of the flat file data.

File schemas are required on a per API per file format basis (for example, you may need to create a positional createOrder schema or a delimited createOrder schema), so you need to write least one unique schema per API you intend to execute. A schema file is written using XML.

The schema file is stored in any location accessible as a CLASSPATH variable. For sample schema files that correspond to the provided sample input files, see the files in the <YFS_HOME>/documentation/code_examples/flatfile/ directory.

Note: XML to XML translation does not require a schema or XSD file.

D.3 Verifying the Text Translator Setup

Before deploying a service, you should create a few test files and run them through the entire process to ensure that all data is being captured and used correctly.

If any errors are encountered while processing a flat file, these errors are logged in two files as specified in the Flat File Receiver configuration file.

- Error file - logs a list of each incident where an error occurs
- Error data file - contains the actual records that are in error

For example, if you have specified that both error files use the default suffix (.err) and you want to process data from an incoming file named text.in, the Text Translator creates error log files when it encounters an error, and thereafter, any additional errors are appended to the end of each of the following files:

- text.in.err - lists the error incidents
- text.in.err.dat - contains the records that are in error

D.3.1 Error Messages

In order to cover a wide variety of situations, error messages typically have more than one parameter. Each parameter points to either field names or field positions. [Table D–1, "Possible Errors in the Error File"](#) lists all possible error conditions. At run-time, the parameters in braces ({}) are filled in with context-specific data.

Table D–1 Possible Errors in the Error File

Error Code	Description
YIF_OVERLAPPING_FIELDS	Field {0} intersects Field {1} in Record {2} This error code indicates that two fields in the same record have overlapping positions
YIF_INVALID_FIELD_BOUNDS	Field {0} has invalid bounds: startPos={1} endPos={2} This error code indicates that the startPosition of a field is greater than the end position of the field
YIF_INVALID_RECORD_TYPE_POS	The position of the record type field is invalid.
YIF_EMPTY_RECORD_TYPE	Record {0} does not have a record type identifier. When the data contains a record whose record type has not been declared in the schema.
YIF_EMPTY_RECORD_FOUND	No fields exist to match for a record. The data contains a record with a record type identifier but the individual fields do not exist
YIF_INVALID_FIELD_GAP	Record {0} should not have a gap [Field {1} endPosition={2}] and [Field {3} startPosition={4}]. In a positional format, two fields should not have a gap. It is fixed width, so unaccounted gaps cannot exist. Fixed width. Period.
YIF_FIELD_OVERLAPS_RECORD_TYPE	In Record {0}, the field {1}[startPos={2}, endPos={3}] overlaps record type field {4} [startPos={5}]. In a positional format, when a field datum overlaps with the boundary of a record type identifier.

Table D–1 Possible Errors in the Error File

Error Code	Description
YIF_FIELD_OVERLAPS_RECORD_TYPE_FIELD	In Record {0}, the field {1} [position={2}] overlaps record type field {3} [position={4}]. In a delimited format, when a field datum overlaps with the boundary of a record type identifier
YIF_SCHEMA_CONTAINS_CYCLE	Schema contains a cycle starting at element: {0} and ending before element: {1}. The schema description does not allow for cyclical containment. If it does, the above error code is reported.
YIF_RECORD_ID_TOO_LARGE	Record Id {0} is too large. Record Identifiers have a fixed width, this error is reported when the size is exceeded.
YIF_FIELD_NOT_DEFINED_IN_RECORD	Field {0} in attribute map does not exist in record {1}. When translating from XML to positional/delimited format, if the field that an XML attribute maps to does not exist.
YIF_TOO_MANY_ATTRIBUTE_MAPS	Number of attribute maps = {0} in elementMap {1} exceeds the number of fields = {2} in record {3}. The number of attributes in an XML element that need to be mapped to fields, cannot exceed the number of fields in the corresponding record.
YIF_INVALID_RID_BOUNDS	RecordIdStartPos: {0} must be less than the RecordIdEndPos: {1}. In a positional format, the bounds of the record identifier must be correct.
YIF_RECORD_ID_NOT_MATCHED	Could not match record id: {0} with any record declared in the schema
YIF_INCORRECT_RECORD_LENGTH	Record?length is incorrect.
YIF_RECORD_LENGTH_TOO_SMALL	Too few fields. Min fields = {0} Found = {1}
YIF_TOO_FEW_FIELDS_FOR_RECORD	Found few fields for record id: {0}. Expected = {1} Found = {2}

Table D–1 Possible Errors in the Error File

Error Code	Description
YIF_TOO_MANY_FIELDS_FOR_RECORD	Too many fields for record id: {0}. Expected = {1} Found = {2}
YIF_RECORD_LENGTH_TOO_LARGE	Too many fields. Max fields = {0} Found = {1}
YIF_RECORD_TOKEN_NOT_FULLY_MATCHED	Scanned record name: {0} is not fully matched. Current Token: {1}
YIF_INCOMPLETE_PREVIOUS_MATCH	Record Set preceding scanned record name: {0} is not completely matched.
YIF_MAX_OCCURENCE_EXCEEDED	Number of matches exceeds Max matches for record: {0}
YIF_FILE_VIOLATES_CONTROL_INFO	File length: {0} must exceed the sum of the fileHeaderLength: {1} and the fileTrailerLength: {2}
YIF_FILE_HEADER_DOES_NOT_EXIST	Expected file header id: {0} is not found
YIF_FILE_TRAILER_DOES_NOT_EXIST	Expected file trailer id: {0} is not found
YIF_FILE_HEADER_DOES_NOT_MATCH	Expected file header id: <{0}> does not match found: <{1}>
YIF_FILE_TRAILER_DOES_NOT_MATCH	Expected file trailer id: <{0}> does not match found: <{1}>
YIF_UNABLE_TO_BOUND_INPUT_STREAM	Unable to set the bounds on input stream ScanLength= <{0}> ScanStartPosition=<{1}>
YIF_DIR_DOES_NOT_EXIST	The specified directory: {0} does not exist.

Table D–1 Possible Errors in the Error File

Error Code	Description
YIF_DIR_IS_NOT_WRITEABLE	The specified directory: {0} does not have write permissions.
YIF_INVALID_INCLUDES_PATTERN	The includes pattern: {0} is not a valid regular expression.

Note: You should periodically check for files within the /error and /complete directories that contain errors or remain unprocessed.

D.4 Text Translator XSD Files

This section describes the essential elements of the Text Translator XSD files. The general fields which are used in all the four files is explained in [Table D–2, "Elements in the Positional Flat File Receiver XSD File"](#).

D.4.1 Text Translator Positional Receiver XSD File

The positional receiver XSD file defines how the data in flat, positional files should be transformed to XML data. [Table D–2, "Elements in the Positional Flat File Receiver XSD File"](#) lists the essential XSD elements and attributes.

Note: "Newline" is the only supported record delimiter used by the Positional Receiver XSD file.

Table D–2 Elements in the Positional Flat File Receiver XSD File

Property	Description
ParserDefaults Element	
RecordIdStartPosition	Required. Integer. This field indicates the start position of the RecordId for each record.
RecordIdEndPosition	Required. Integer. This field indicates the end position of the RecordId for each record.

Table D–2 Elements in the Positional Flat File Receiver XSD File

Property	Description
DefaultRecordDelimiter	Optional. The default record delimiter is Newline.
DefaultEscapeCharacter	Optional. CharacterType. The default escape character is \.
DefaultPadCharacter	<p>Required. String. Minimum length=1. Maximum length=1. Fills the non-data portion of a field with any single character. Valid values include a space or zero as well as any other character. The default value is #.</p> <p>For example if you specify trailing ###'s in a field they will be ignored.</p> <p>This character is used for visual convenience when you want to space out the fields correctly and do not want to rely on the blank space.</p>
SkipCarriageReturn	Optional. Boolean. Defaults to true.
<p>FileHeader Element</p> <p>The header that is written to a file before writing anything else. This can be used as control information. Each transaction set starts with a header record. If it contain internationalized text, its length must be byte accounted.</p> <p>For example, a Sales Order record set has an OrderHeader record that specifies the beginning of the transaction set.</p> <p>This element is optional. However, if FileHeader element is provided the FileTrailer element must also be present. These two elements are used to include text at the beginning and end of each file respectively.</p>	
FileHeaderName	Optional. String. The descriptive name of the FileHeader. However, this is not used for processing.
FileHeaderId	Required. String. This attribute provides the text to match, at the start of each file.
FileHeaderStartPos	Required. Integer. The starting position of the header.
FileHeaderLength	Required. Integer. The length of the header.

Table D–2 Elements in the Positional Flat File Receiver XSD File

Property	Description
FileTrailer Element	
The trailer that is written to the end of a file. This can be used as control information to verify if a file is indeed complete. If it contain internationalized text, its length must be byte accounted.	
FileTrailerName	Optional. String. The descriptive name of the FileTrailer. However, this is not used for processing.
FileTrailerId	Required. String. This attribute provides the text to match, at the end of each file.
FileTrailerStartPos	Required. Integer. The starting position of the trailer.
FileTrailerLength	Required. Integer. The length of the trailer.
CharacterType Element	
This element specifies the character type of all the elements and attributes used in the XSD file. Required. String. Minimum length=1. Maximum length=1.	
Root Element	
The elements and attributes defined under <code>Root</code> elements portrays the organization of the input flat file.	
Name	Required. NMTOKEN. The name of the root element. This is the same root-element name of the XML you are building.
Description	Optional. String. The description of the root element. This attribute is not used for processing.
XMLName	Optional. Name of the root entity in the translated XML.
Header Element	
Required. This is the first record that is read under <code>Root</code> element.	
Name	Required. NMTOKEN. The name of the header element. This is the tag name in your XML file.

Table D–2 Elements in the Positional Flat File Receiver XSD File

Property	Description
RecordName	Required. NMTOKEN. The name of the record. This field must match the Name attribute of the Record element. This name is the identifier for the RecordName used in the flatfile.
MinOccurence	Optional. Integer. Minimum number of times this sequence can occur. By default, this sequence should occur at least once. A value of 0 means that the occurrence of this sequence is optional. Defaults to 1.
MaxOccurence	Required. Integer. The maximum number of times this sequence can occur. A value of 0 means that this sequence can occur an unlimited number of times. Defaults to 1.
Terminal Element	
Defines a record that is not a part of a sequence or a choice entity. A terminal entity is a leaf node in the hierarchy.	
MinOccurence	Optional. Integer. Minimum number of times this sequence can occur. By default, this sequence should occur at least once. A value of 0 means that the occurrence of this sequence is optional. Defaults to 1.
MaxOccurence	Optional. Integer. The maximum number of times this sequence can occur. A value of 0 means that this sequence can occur an unlimited number of times. Defaults to 1.
Name	Required. NMTOKEN. Name of the terminal node.
RecordName	Required. NMTOKEN. Name of the record that corresponds with this terminal node.

Table D–2 Elements in the Positional Flat File Receiver XSD File

Property	Description
<p>Sequence Element</p> <p>This element is required and is of type SequenceType.</p> <p>The record corresponding to a sequence entity which indicates the beginning of a sequence. This record may have sub-record of multiple types like, Terminal, Sequence or Choice elements.</p> <p>For example, an order transaction could contain an orderline, and an orderline could contain a sequence of one or more line items. In this case, the orderline record corresponds to a sequence entity, and this sequence entity contains another sequence entity corresponding to a line item as a child.</p>	
MinOccurence	Optional. Integer. Minimum number of times this sequence can occur. By default, this sequence should occur at least once. A value of 0 means that the occurrence of this sequence is optional. Defaults to 1.
MaxOccurence	Optional. Integer. The maximum number of times this sequence can occur. A value of 0 means that this sequence can occur an unlimited number of times. Defaults to 1.
Name	Required. NMTOKEN. The name of the sequence element.
RecordName	Required. NMTOKEN. The record name of the sequence element identified in the flatfile.
<p>Choice Element</p> <p>This element is required and is of ChoiceType.</p> <p>The choice entity declaration defines one entity in a group of child elements that appears in the data. The choice entity does not correspond to a record. It is simply a grouping of a record's child elements, specifying that exactly one of its child elements can occur. However, each child element can correspond to a sequence element or a terminal element.</p>	

Table D–2 Elements in the Positional Flat File Receiver XSD File

Property	Description
MinOccurrence	Optional. Integer. Minimum number of times this sequence can occur. By default, this sequence should occur at least once. A value of 0 means that the occurrence of this sequence is optional. Defaults to 1.
MaxOccurrence	Optional. Integer. The maximum number of times this sequence can occur. A value of 0 means that this sequence can occur an unlimited number of times. Defaults to 1.
Name	Required. NMTOKEN. The name of the choice element.
RecordDefinitions\Record Element	
A record describes a line in the flat file. This record definition is translated into an XML element.	
RecordId	Required. NMTOKEN. This is the RecordId in the source XML file.
Name	The name of the record. This is used to associate with a Header, Terminal, Sequence or Choice Name attribute.
Description	String. The description of the name. This is not used while processing the file.
XMLName	Required. NMTOKEN. The tag name of the output element.
Field Element	
Each record consists of fields, which are translated into attributes or child elements depending on the ContainmentType.	
Name	Required. NMTOKEN. The name of the field. This name must be unique within a record.
XMLName	Required. NMTOKEN. The output attribute or element name of the XML.
Description	Optional. String. The description on the field.

Table D-2 Elements in the Positional Flat File Receiver XSD File

Property	Description
ContainmentType	This is either an Attribute or Element in the XML. If it is an Attribute, a new attribute is set to this field's value. If it is an Element, a new child element will be created with the tag name set to XMLName and the value set to value of this field. Defaults to Attribute.
StartPosition	Required. Integer. The StartPosition should be one number greater than the EndPosition of the previous record or field, so that these two fields or records are contiguous.
EndPosition	Required. Integer. The ending position of the field.
fileLayoutType Element	Required. String. Values are Positional - flat file with fixed-length fields Delimited - flat file with varying-length fields XML - flat file with fields denoted by XML tags
recordLayoutType Element	Required. String. Values are: Positional Delimited
Justification	Optional. Specifies the alignment of data. Right - Aligns data to the right. Left - Aligns data to the left when the data is less than the maximum field length. This also aligns data to the left when the amount of data is less than the minimum length requirement. Default.
DefaultValue	Optional. NMTOKEN.
PadCharacter	Required. String. Minimum length=1. Maximum length=1. Fills the non-data portion of a field with any single character. Valid values include a space or zero.
PadCharacterType Element	

Table D–2 Elements in the Positional Flat File Receiver XSD File

Property	Description
	Required. String. Minimum length=1. Maximum length=1. Fills the non-data portion of a field with any single character. Valid values include a space or zero.
JustificationType Element	
	Optional. Specifies the alignment of data. Right - Aligns data to the right. Left - Aligns data to the left when the data is less than the maximum field length. This also aligns data to the left when the amount of data is less than the minimum length requirement. Default.

D.4.2 Text Translator Delimited Receiver XSD File

The Text Translator delimited receiver XSD file defines how the data in delimited flat files should be transformed to XML data. [Table D–3, "Elements in the Delimited Flat File receiver XSD File"](#) defines the essential elements and attributes.

The `Root`, `FileHeader` and `FileTrailer` element definitions remain the same as explained in [Section D.4.1, "Text Translator Positional Receiver XSD File"](#).

Table D–3 Elements in the Delimited Flat File receiver XSD File

Property	Description
ParserDefaults Element	
RecordIdStartPosition	Required. Integer. Indicates the start position of a record.
DefaultRecordDelimiter	Required. RecordDelimiterType. The default delimiter between records is <code>Newline</code> .
DefaultFieldDelimiter	Optional. CharacterType. The default field delimiter between the fields is comma <code>[,]</code> .
DefaultEscapeCharacter	Optional. CharacterType. The default escape character is <code>\</code> .

Table D-3 Elements in the Delimited Flat File receiver XSD File

Property	Description
DefaultPadCharacter	<p>Required. String. Minimum length=1. Maximum length=1. Fills the non-data portion of a field with any single character. Valid values include a space or zero as well as any other character. The default value is #.</p> <p>For example if you specify trailing ###'s in a field they will be ignored.</p> <p>This character is used for visual convenience when you want to space out the fields correctly and do not want to rely on the blank space.</p>
DefaultWrapCharacter	<p>The Wrapping/Quote character is parameterized. One can define the default wrapping character in the Flat file-Delimited-Schema (Service component specific).</p> <p>A sample entry is given below:</p> <pre data-bbox="658 812 1315 1315"> <?xml version="1.0" encoding="UTF-8"?> <FlatfileDelimitedSchema xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation='delimitedreceiver.xsd'> <ParserDefaults DefaultRecordDelimiter="Newline" DefaultFieldDelimiter=" " DefaultPadCharacter="#" RecordIdStartPosition="1" DefaultWrapCharacter="@"> ... </ParserDefaults> </FlatfileDelimitedSchema> </pre> <p>In the above example, the Wrap/Quote character will be set to <code>_@_</code>.</p> <p>Note: The default wrap character attribute is optional, and by default has a value of <code>""</code>. If <code>DefaultWrapCharacter="NONE"</code>, then no wrap character is used.</p>

Table D-3 Elements in the Delimited Flat File receiver XSD File

Property	Description
SkipCarriageReturn	Optional. Boolean. Defaults to true.
TranslationProperties Attributes	
SchemaXMLFile	Required. String. the relative path to the XSD description file of the translation from XML input to a flat file. The location is relative to the CLASSPATH of the FileSendAgent. For example, if the flat file XSD XML file is located in the <YFS_HOME>/bin/test.xml file, and <YFS_HOME> is in the CLASSPATH, then the value of the attribute is /bin/test.xml file.
RecordDefinitions\Record Element	
A record describes a line in the flat file. This record definition is translated into an XML element.	
RecordId	Required. NMTOKEN. This is the RecordId in the source XML file.
Name	The name of the record. This is used to associate with a Header, Terminal, Sequence or Choice Name attribute.
Description	String. The description of the name. This is not used while processing the file.
XMLName	Required. NMTOKEN. The tag name of the output element.
Field Element	
Each record consists of fields, which are translated into attributes or child elements depending on the ContainmentType.	
Name	Required. NMTOKEN. The name of the field. This name must be unique within a record.
XMLName	Required. NMTOKEN. The output attribute or element name of the XML.

Table D–3 Elements in the Delimited Flat File receiver XSD File

Property	Description
FieldPosition	Required. Integer The position of this field within the record. If the RecordId is at position 1, then the numbering of fields should begin at position 2.
ContainmentType	This is either an Attribute or Element in the XML. If it is an Attribute, a new attribute is set to this field's value. If it is an Element, a new child element will be created with the tag name set to XMLName and the value set to value of this field. The default value is Attribute.

D.4.3 Text Translator Positional Sender XSD File

The Text Translator positional sender XSD file defines how XML data should be transformed into a flat, positional file. [Table D–4, "Elements in the Positional Flat File Sender XSD File"](#) lists the essential XSD elements and attributes.

The `FileHeader` and `FileTrailer` element definitions remain the same as explained in [Section D.4.1, "Text Translator Positional Receiver XSD File"](#).

Table D–4 Elements in the Positional Flat File Sender XSD File

Property	Description
ParserDefaults Element	
RecordIdStartPosition	Required. Integer. This field indicates the start position of the RecordId for each record.
RecordIdEndPosition	Required. Integer. This field indicates the end position of the RecordId for each record.
DefaultRecordDelimiter	Required. RecordDelimiterType. The default delimiter between records is <code>Newline</code> .
DefaultEscapeCharacter	Optional. CharacterType. The default escape character is <code>\</code> .

Table D-4 Elements in the Positional Flat File Sender XSD File

Property	Description
DefaultPadCharacter	<p>Required. String. Minimum length=1. Maximum length=1. Fills the non-data portion of a field with any single character. Valid values include a space or zero as well as any other character. The default value is #.</p> <p>For example if you specify trailing ###'s in a field they will be ignored.</p> <p>This character is used for visual convenience when you want to space out the fields correctly and do not want to rely on the blank space.</p>
SkipCarriageReturn	Optional. Boolean. Defaults to true.
ElementMapList Element	
This element maps the elements in the XML to the records in the flat file.	
ElementMap Element	
ElementName	Required. NMTOKEN. The name of the element in the XML file.
RecordId	Required. NMTOKEN. The record identifier to write.
AttributeMap Element	
AttributeName	Required. NMTOKEN. The name of the attribute.
FieldName	Required. NMTOKEN. The field corresponding to the attribute. This field corresponds to the Name attribute of the Field element.
TruncateData	Optional. Boolean. Defaults to true.
RecordDefinitions\Record Element	
A record describes a line in the flat file. This record definition is translated into an XML element.	
RecordId	Required. NMTOKEN. This is the RecordId in the source XML file.

Table D-4 Elements in the Positional Flat File Sender XSD File

Property	Description
Name	The name of the record. This is used to associate with a Header, Terminal, Sequence or Choice Name attribute.
Description	String. The description of the name. This is not used while processing the file.
XMLName	Required. NMTOKEN. The tag name of the output element.
Field Element	
Each record consists of fields, which are translated into attributes or child elements depending on the <code>ContainmentType</code> .	
Name	Required. NMTOKEN. The name of the field which is used as a reference in the <code>AttributeMap</code> . This name must be unique within a record.
XMLName	Required. NMTOKEN. The output attribute or element name of the XML.
StartPosition	Required. Integer. The StartPosition should be one number greater than the EndPosition of the previous record or field, so that these two fields or records are contiguous.
EndPosition	Required. Integer. The ending position of the field.
PadCharacter	Required. String. Minimum length=1. Maximum length=1. Fills the non-data portion of a field with any single character. Valid values include a space or zero.
Justification	Optional. Specifies the alignment of data. Right - Aligns data to the right. Left - Aligns data to the left when the data is less than the maximum field length. This also aligns data to the left when the amount of data is less than the minimum length requirement. Default.

D.4.4 Text Translator Delimited Sender XSD File

The Text Translator delimited Sender XSD file defines how to transform XML data to delimited files. [Table D–5, "Elements in the Delimited Flat File Sender XSD File"](#) lists the essential elements and attributes.

The `Root`, `FileHeader` and `FileTrailer` element definitions remain the same as explained in [Section D.4.1, "Text Translator Positional Receiver XSD File"](#).

Table D–5 Elements in the Delimited Flat File Sender XSD File

Property	Description
ParserDefaults Element	
RecordIdStartPosition	Required. Integer. Indicates the start position of a record.
DefaultRecordDelimiter	Required. RecordDelimiterType. The default delimiter between records is <code>NewLine</code> .
DefaultFieldDelimiter	Optional. CharacterType. The default field delimiter between the fields is comma <code>[,]</code> .
DefaultEscapeCharacter	Optional. CharacterType. The default escape character is <code>\</code> .
DefaultPadCharacter	Required. String. Minimum length=1. Maximum length=1. Fills the non-data portion of a field with any single character. Valid values include a space or zero as well as any other character. The default value is <code>#</code> . For example if you specify trailing <code>###</code> 's in a field they will be ignored. This character is used for visual convenience when you want to space out the fields correctly and do not want to rely on the blank space.
SkipCarriageReturn	Optional. Boolean. Defaults to <code>true</code> .
SuppressEORFieldDelimiter	Optional. Boolean. Defaults to <code>False</code> . This attribute when passed as <code>True</code> does not print the delimiter at the end of the record.

Table D–5 Elements in the Delimited Flat File Sender XSD File

Property	Description
ElementMapList Element	
This element maps the elements in the XML to the records in the flat file.	
ElementMap Element	
ElementName	Required. NMTOKEN. The name of the element in the XML file.
RecordId	Required. NMTOKEN. The record identifier to write.
AttributeMap Element	
AttributeName	Required. NMTOKEN. The name of the attribute.
FieldName	Required. NMTOKEN. The field corresponding to the attribute. This field corresponds to the Name attribute of the Field element.
TruncateData	Optional. Boolean. Defaults to true.
RecordDefinitions\Record Element	
A record describes a line in the flat file. This record definition is translated into an XML element.	
RecordId	Required. NMTOKEN. This is the RecordId in the source XML file.
Name	The name of the record. This is used to associate with a Header, Terminal, Sequence or Choice Name attribute.
Description	String. The description of the name. This is not used while processing the file.
XMLName	Required. NMTOKEN. The tag name of the output element.
WriteRecordId	The default value is Y. If set to N, no record ID is written to the output file.

Table D–5 Elements in the Delimited Flat File Sender XSD File

Property	Description
Field Element	
Each record consists of fields, which are translated into attributes or child elements depending on the <code>ContainmentType</code> .	
Name	Required. NMTOKEN. The name of the field which is used as a reference in the <code>AttributeMap</code> . This name must be unique within a record.
XMLName	Required. NMTOKEN. The output attribute or element name of the XML.
FieldPosition	Required. Integer The position of this field within the record. If the <code>RecordId</code> is at position 1, then the numbering of fields should begin at position 2.

D.5 Running the Text Translator

The Text Translator uses scripts located in the `<YFS_HOME>/bin/` directory.

D.5.1 Specifying the Text Translator Startup Script

If you want to ensure that the file schemas and XSDs are parsed correctly and understood, you can add the following line to the startup script:

```
"-Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryImpl
1
-Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBuilder
rFactoryImpl"
```

For example, the entire Java line in the `startIntegrationServer.sh` script might resemble the following example:

```
java
-Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryImpl
-Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBuilder
rFactoryImpl com.yantra.integration.adapter.IntegrationAdapter TextTranslatorJMS
```

E

Document Types

This appendix provides a reference for the different document types used by Yantra 7x.

Table E-1 Document Types

Document Type	Number	Description
Order		
Sales Order	0001	This document type is used to sell items or services, either from business to business or business to customer.
Planned Order	0002	This document type is used to plan an order that will take place in the future.
Return Order	0003	This document type is used for returning items to the seller.
Template Order	0004	This document type is used to create a template that future orders can be modeled from.
Purchase Order	0005	This document type is used by a business to purchase items or services from another business.
Transfer Order	0006	This document type is used to transfer items from one organization to another (for example, a warehouse to a distribution center, a warehouse to another warhouse, a distribution center to a store).
Load		

Table E-1 Document Types

Document Type	Number	Description
Load	1001	This document type is used for a delivery plan which consolidates multiple shipments into a single delivery.
General		
General	2001	This document type is used for transactions and services that do not fall under any other document types.
Putaway	2002	This document type is used for transactions and services that are used for the putaway related processes (to specify the location where the products coming into the warehouse would be putaway).
Layout	2003	This document type is used for transactions and services that are used for configuring the warehouse layout and related processes.
Inventory	2004	This document type is used for transactions and services that are used for inventory tracking, maintenance, and related processes.
Trailer	2005	This document type is used for transactions and services that are used for the trailer loading and unloading related processes.
Task	2006	This document type is used for transactions and services that plan all tasks that need to be done within a warehouse.
Move Request	2007	This document type is used for transactions and services that describe how products should be moved from one location to another within a warehouse.
Manifest	2008	This document type is used for transactions and services that relate to the manifest process, and provide the carrier that describes what is in the shipment.

Table E-1 Document Types

Document Type	Number	Description
Over Pack	2009	This document type is used for transactions and services that relate to the overpacking process, which involves packing one or more outbound containers into another outbound container.
Count		
Count	3001	This document type is used for transactions and services that relate to performing a physical count of inventory in the warehouse.
Container		
Container	5001	This document type is used for transactions and services that relate to containers used to pack outbound goods.
Wave		
Outbound Picking	4001	This document type is used for transactions and services that relate to the outbound picking process.
Work Order		
Work Order	7001	<p>This document type is used in two different contexts:</p> <ul style="list-style-type: none"> • WMS: Tasks that need to be done in the warehouse, typically against items (tagging, kitting, etc.) • DOM: When a Provided Service is ordered, a Work Order gets created, typically for tasks like installation and repairs.

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](#)

F.1 Sales Order

F.1.1 Order Fulfillment

Table F-1 Order Fulfillment Condition Builder Attributes

Attribute	Description
Order Attributes	
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.
Participant Attributes	
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 F–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.
Item Attributes	
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.
Sourcing Attributes	
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.
Related Order Attributes	
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 F–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}	A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i> . Note: 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.

F.1.2 Order Negotiation

Table F–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 F–2 Order Negotiation Condition Builder Attributes

Attribute	Description
Payment Terms	The payment terms for this order.
{Enter Your Own Attribute}	A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i> . Note: 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.

F.1.3 Outbound Shipment

Table F–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 F–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}	A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i> . Note: 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.

F.2 Planned Order

F.2.1 Planned Order Execution

The Planned Order Execution condition builder attributes are identical to the [Order Fulfillment](#) attributes.

F.2.2 Planned Order Negotiation

The Planned Order Negotiation condition builder attributes are identical to the [Order Negotiation](#) attributes.

F.3 Return Order

F.3.1 Reverse Logistics

Table F–4 Return Fulfillment Condition Builder Attributes

Attribute	Description
Order Attributes	
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.
Participant Attributes	
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 F-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.
Item Attributes	
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.
Sourcing Attributes	
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.
Related Order Attributes	
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 F–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}	A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i> . Note: 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.

F.3.2 Return Shipment

The Return Shipment condition builder attributes are identical to the [Outbound Shipment](#) attributes.

F.3.3 Return Receipt

Table F–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 F–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}	A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i> . Note: 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.

F.4 Template Order

The Template Order condition builder attributes are identical to the [Order Fulfillment](#) attributes.

F.5 Purchase Order

F.5.1 Purchase Order Execution

Table F–6 Purchase Order Execution Condition Builder Attributes

Attribute	Description
Order Attributes	
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 F–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.
Participant Attributes	
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.
Item Attributes	

Table F-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.
Sourcing Attributes	
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.
Related Order Attributes	
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>Note: 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>

F.5.2 Purchase Order Negotiation

Table F–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>Note: 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>

F.5.3 Inbound Shipment

The Inbound Shipment condition builder attributes are identical to the [Outbound Shipment](#) attributes.

F.5.4 Purchase Order Receipt

The Purchase Order Receipt condition builder attributes are identical to the [Return Receipt](#) attributes.

F.6 Transfer Order

F.6.1 Transfer Order Execution

The Transfer Order Execution condition builder attributes are identical to the [Order Fulfillment](#) attributes.

F.6.2 Transfer Order Delivery

The Transfer Order Delivery condition builder attributes are identical to the [Outbound Shipment](#) attributes.

F.6.3 Transfer Order Receipt

The Transfer Order Receipt condition builder attributes are identical to the [Return Receipt](#) attributes.

F.7 Load Execution

Table F–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 F–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}	A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i> . Note: 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.

F.8 General

Table F–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 F–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}	A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i> . Note: 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.

F.9 WMS Putaway

The WMS Putaway condition builder attributes are identical to the [General](#) attributes.

F.10 WMS Layout Definition

The WMS Layout Definition condition builder attributes are identical to the [General](#) attributes.

F.11 WMS Inventory

The WMS Layout Inventory condition builder attributes are identical to the [General](#) attributes.

F.12 Trailer Loading

The Trailer Loading condition builder attributes are identical to the [General](#) attributes.

F.13 Task Execution

The Task Execution condition builder attributes are identical to the [General](#) attributes.

F.14 Move Request Execution

The Move Request Execution condition builder attributes are identical to the [General](#) attributes.

F.15 Manifesting

The Manifesting condition builder attributes are identical to the [General](#) attributes.

F.16 Over Pack Build

The Over Pack Build condition builder attributes are identical to the [General](#) attributes.

F.17 Count Execution

Table F–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 F–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 F–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}	A customizable condition builder attribute. For more information on customizing this field, refer to the <i>Yantra 7x Customization Guide</i> . Note: 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.

F.18 Pack Process

Table F–11 Pack Process Condition Builder Attributes

Attribute	Description
Node Attributes	
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 F–11 Pack Process Condition Builder Attributes

Attribute	Description
Ship to Country	The code of the country to which the shipment is being shipped.
Organization Attributes	
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.
Shipment Attributes	
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.
Container Attributes	
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 F–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>Note: 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>

F.19 Outbound Picking

Table F–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>Note: 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>

F.20 VAS Process

Table F–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 F–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>Note: 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

Abort Processing on Error field, 343
Account Number With Hub field, 32, 116
Action Code field, 203
Action Group field, 203
Action Name field, 86, 203
actions, 201
 creating, 202
 deleting, 207
 modifying, 207
Active field, 231
adapter nodes, 208, 210
Administered By field, 34
Agent Criteria Group field, 84
Agent Server field, 167
Alert Console template, 205
alert nodes, 610
Alert Queue Name field, 167, 607
alert queues, 345
 based on size, 347
 creating, 345
 deleting, 350
 modifying, 350
 unassigned alerts, 348
 unresolved alerts, 349
 user list, 350
Alert Type field, 607
alerts, 345
Allow Chained Order Creation field, 143
Allow Demand Updates For The Seller Organization field, 142
Allow Inventory Check During Schedule And Release field, 142

Allow Invoice Creation field, 142
Allow Price Calculation For Confirmed Orders field, 143
Allow Price Calculation For Draft Orders field, 143
Allow Pro Forma Invoice Creation For Shipments flag, 143
Allow Propagation Of Changes To Derived Parent field, 144
Allow Shipping Of Hazardous Materials field, 75, 78, 82
Allow Supply Updates For The Buyer Organization field, 142
API Name field, 311
API nodes, 613
Application menu
 modifying, 298
application rules side panel, 10
Apply SCM on all cases field, 64
Apply SCM on all shipping units field, 64
Appointment Required field, 49
asynchronous transport nodes, 209
Attribute Location field, 342
Audit Queue field, 347
Authority Type field, 116
Available field, 233

B

backwards compatibility
 order hold deprecated functionality, 253
bar codes, 249
Base Currency field, 291
base document types, 131, 132

- base repositories, 147
- Base Transaction Name field, 163
- base transactions, 158
- Batch Sheet for picking, 377
- Billing Address field, 232
- Binding field, 315
- BOL Prefix field, 84
- Break Bulk Node field, 59
- business communication components, 5, 327
 - business document codes, 331
 - document format codes, 329
 - protocol codes, 327
- business document codes
 - creating, 331
 - deleting, 332
 - modifying, 332
- business documents
 - classifying, 332
- business models, 2
- buyer documents, 332
- Buyer Instructions
 - Buyer Instruction Type Field, 72
- buyers, 3, 38
 - packaging parameters, 62
 - print format preferences, 69

C

- Calendar ID field, 120
- calendar inheritance, 118
 - child calendar, 118
 - limitations, 119
 - parent calendar, 118
- calendars
 - creating an exception, 123
 - defining an organization's, 118
 - defining defaults, 122
 - inheritance See also calendar inheritance, 118
- Call DB Extension field, 206
- Call HTTP Extension field, 206
- Call Java Extension field, 205
- Can Enterprise Configure Event Handler field, 176
- capacity consolidation level, 251
- Capacity Organization Is field, 113
- capacity organizations, 112

- CAPACITYPRG purge rule, 242
- carrier documents, 332
- Carrier field, 98
- Carrier Service Code field, 59
- carrier services, 73
- carriers, 38
 - defining attributes, 73
 - defining less-than truckload carrier services, 76
 - defining parcel carrier services, 79
 - defining truckload carrier services, 73
 - roles
 - carriers, 38
- Carrier/Service field, 59, 77
- Cart Manifest for picking, 377
- Catalog Defined By field, 111
- catalog model, 251
- catalog organization, 111
- catalogs, 111
- Chained Document Type field, 163
- Chained Drop Ship Order Document Type field, 144
- chained orders, 93, 143
- Chained Procurement Purchase Order Document Type field, 144
- Chained Procurement Transfer Order Document Type field, 144
- Check Digit Determination field, 100
- Check for Prior Exception field, 608
- child organizations
 - viewing, 117
- Class Name field, 196, 205, 338
- Close Window On Complete field, 316
- Code Description field, 337
- Code Name field, 338
- Code Value API field, 338
- commoncodes, 320
- communication, 5
- communication protocols, 113
 - creating, 113
 - deleting, 115
 - modifying, 114
- Compliance Service field, 68
- compliance services, 65
 - creating, 66

- deleting, 69
 - modifying, 69
- component nodes, 208, 210, 609
- composite service nodes, 615
- condition builder, 661 to 683
 - using, 197
- Condition Group field, 196
- condition groups, 196
- Condition ID field, 196
- Condition Name field, 196
- condition nodes, 616
- Condition Properties field, 197
- Condition Value field, 197
- conditions, 194
 - creating, 195
 - deleting, 199
 - modifying, 198
 - viewing affected entities, 199
- 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
- Configuring, 31
- Connector node, 208
- console actions, 314
- console APIs, 310
- console entities, 305
- console icons, 317
- console inner panels, 313
- console links, 316
- console list views, 309
- console related entities, 306
- console search view, 307
- Consolidate New Order Releases field, 144
- Consolidate upto volume threshold of field, 51
- Consolidate upto weight threshold of field, 50
- Consolidator field, 59, 60
- Constant field, 343
- Contact Address field, 232
- Conversion Rate From Base field, 291
- Conversion Rate To Base field, 291
- Corporate Address field, 35
- Cost Factor Group to be Used for Landed Cost
 - Calculations field, 45, 47
- Cost Factor Group to be Used for Physical Kit Cost
 - Calculations field, 45
- Cost Factor Group to be Used for Standard Cost
 - Calculations field, 45, 47
- cost factor preferences, 44, 47
- Cost Factor Preferences tab, 44, 47
- count execution process type, 133
- country
 - locale, 295
- country codes
 - creating, 278
 - deleting, 279
 - modifying, 279
- Country field, 58, 359
- Create Reservation On Order Creation field, 142
- Create Shipments For Products Included In Work
 - Order field, 144
- currency conversions, 290
 - creating, 290
 - deleting, 292
 - modifying, 291
- currency definitions, 287
 - creating, 288
 - deleting, 289
 - modifying, 289
- Currency field, 295
- custom common code types, 320
 - creating, 320
 - deleting, 322
 - modifying, 321
- custom common code values
 - deleting, 324
 - modifying, 323
- custom common codes
 - adding values to, 322

- custom error codes, 324
 - adding, 325
 - deleting, 326
 - modifying, 325
 - searching, 324
- Customer PO # field, 49, 64, 65

D

- data security component, 626
 - custom validation, 627
 - enterprise validation, 627
- Data Security Group field, 232
- data security groups, 232, 237
 - configuring component, 238
 - creating, 238
 - deleting, 240
 - modifying, 240
- database nodes, 560
 - database receiver properties, 562
 - database sender properties, 560
- database receiver properties, 562
- database sender properties, 560
- Date Format field, 296
- date formats
 - creating, 282
 - deleting, 283
 - modifying, 282
- Date Time Format field, 296
- Date/Hour/Minute Format field, 296
- dates
 - calculations, 87
 - Schedule date calculations, 87
- date/time formats, 285
 - creating, 285
 - deleting, 287
 - modifying, 286
- Day Display Date Format field, 296
- DCS 6.2 database node, 565
 - defining defaults, 565
 - error processing, 565
 - interface tables, 565
 - sender, 565
- DCS 6.2 database nodes
 - DCS 6.2 database receiver properties, 566
 - DCS 6.2 database sender properties, 565
 - receiver, 566
- DCS 6.2 database receiver properties, 566
- DCS 6.2 database sender properties, 565
- Default Declared Value field, 85
- Default Delivery Method Based On Catalog field, 141
- Default Document Type Access field, 239
- Default Effective From field, 120
- Default Effective To field, 120
- Default Enterprise Access field, 239
- Default Label Format, 380
- Default No of Copies field, 44, 47, 70
- defaulting component, 624
 - configuration properties, 625
- Delay shipment by not more than field, 50
- Demand Type for Release field, 142
- Demand Type For Schedule field, 142
- demand types, 142
- Department Abbreviation field, 126
- Department Code field, 49, 64, 65
- departments
 - creating, 125
 - deleting, 126
 - modifying, 126
 - viewing, 125
- Derived Document Type field, 163
- derived transactions
 - creating, 185
- detail view components
 - anchor pages
 - defining, 308
- device Attributes, 370
- device attributes, 372
- Device ID, 370
- device sub types, 370
 - creating, 366
 - defining, 366
 - deleting, 368
 - modifying, 367
- device types, 365, 370
 - creating, 364
 - defining, 363
 - deleting, 366
 - modifying, 365

- devices
 - creating, 369
 - creating new from existing, 373
 - defining, 363, 368
 - deleting, 376
 - device attributes
 - setting up, 372
 - modifying, 375
- dimension unit of measure
 - creating, 262, 264, 266
 - creating conversion rate, 262, 265, 267
 - deleting, 263, 266, 268
 - deleting conversion rate, 263, 265, 268
 - modifying, 262, 264, 267
 - modifying conversion rate, 263, 265, 268
- Dimension UOM field, 295
- Display Error Details field, 226
- Distance Per Day field, 75, 78, 82
- Do not apply SCM field, 64
- Do Not Generate field, 98
- document definitions, 37
- Document Description, 379
- document format codes
 - creating, 329
 - deleting, 330
 - modifying, 330
- document templates, 145
- Document Type field, 136, 184, 247, 306
- document types, 132, 244
 - configuring, 131
 - modifying description, 136
 - saving as new, 134
- DocumentType attribute, 306
- draft orders, 143
- Driver Date field, 140
- drop ship, 144
- drop statuses, 189
 - adding, 180
 - deleting, 181
- Drop Statuses field, 164
- DUNS Number field, 32, 34

E

Economic shipping parameters maintained field, 50

- Effective Date field, 54
- Effective From field, 291
- Effective Period field, 122
- Effective To field, 291
- EJB nodes, 569
- Electronic Code field, 74, 77, 81
- E-mail nodes, 617
- E-mail Protocol field, 252
- E-mail Server IP Address field, 252
- E-mail Server Listener Port field, 252
- E-mail Server Name field, 252
- E-mail template, 205
- end nodes, 211
- End Time field, 120, 123
- Enterprise Code field, 67
- enterprises, 2, 3, 37
 - defining attributes, 40
 - adding organizations, 41
 - modifying associated organizations, 42
 - removing associated organizations, 43
 - print format preferences, 43
- Entity Key Name field, 314
- error codes
 - customizing, 324
- Escalate Alert After field, 348, 349
- Escalate If Number of Unresolved Alert Exceeds field, 348
- Euro currency, 287
- Euro currency membership, 287
- event handlers
 - defining, 177
- Event ID field, 176
- Event Name field, 176
- events, 164
 - adding to a transaction, 175
 - deleting, 177
 - modifying, 176
- Events field, 164
- Exception Group field, 608
- exception handling, 560, 607
- Exception Reference Name field, 608
- Exception Reference Value field, 609
- Exception References tab, 608
- Exception tab, 606
- exchange rates, 290

Execute field, 205
Exemption Certificate field, 116
Expedite shipment by not more than field, 50
Expiration Date field, 54
Export License Number field, 84
EXPORTTBLPRG purge rule, 242
Extended Ship Mode field, 74, 78, 81
extended statuses
 creating, 190
 deleting, 192
 modifying, 191
extended transactions, 160
External Application field, 85
Externally Generated field, 98
Externally-triggered transactions
 defining, 164
externally-triggered transactions, 159

F

File IO nodes, 571
File IO receiver node properties, 572
File IO sender nodes, 571
File transfer protocol. See FTP
Fixed Transit Days field, 75, 78, 82
Freight Terms field, 55
From field, 59
From Node field, 94
From Organization field, 341
From System field, 341
FTP nodes, 578
FTP receiver properties, 581
FTP sender properties, 579
FTP (File Transfer Protocol), 578

G

general base document type, 132
general process type, 133
Get Capacity Real Time field, 113
Get External Inventory Real Time flag, 105
Get External Supply Real Time flag, 106, 109
Get Inventory Real Time field, 109
GLN field, 84
Group Description field, 222

Group ID field, 222, 238
Group Name field, 222

H

Has Freezer field, 75, 78, 82
Hide Maximum Records Field field, 308
Hide Navigation Panel field, 309
Hide Title Bar field, 309
holds
 putting on, 253
 This Transaction Can Be Stopped From
 Processing Orders That Are On Hold, 164
Host Code field, 74, 78, 81
Hour/Minute Time Format field, 296
HP LaserJet 5P, 366
HTTP nodes, 584
Hub, 36, 148

I

Icon field, 300
Identified By Parent As field, 84
Ignore Default API checkbox field, 310
Ignore Default API field, 308, 309
Implement As A Java Class field, 185, 247
Implement As A Service field, 184, 247
IMPORTTBLPRG purge rule, 242
In a Synchronous Mode field, 213
In an Asynchronous Mode field, 213
inbound shipment process type, 133
INBOXPRG purge rule, 242
Inherit Calendar definition field, 120
inheritance
 determining, 11
initial context factory codes, 256
 creating, 256
 deleting, 258
 modifying, 257
Initial Context Factory field, 168
Input Key Name field, 316
Input Name Space field, 315
installation rules, 248
internationalization rules, 277
 country codes, 277

- currency conversions, 290
- currency definitions, 287
- date formats, 281
- date/time formats, 285
- language codes, 279
- time formats, 283
- inventory consolidation levels, 250
- inventory costing factors, 44, 47
- Inventory Information Not Available To Yantra field, 105, 108
- Inventory Is Made Available To Yantra field, 105, 108
- Inventory Maintained Within Yantra field, 105, 108
- Inventory Organization Is field, 109
- inventory organizations, 108, 109
- INVENTORYPRG purge rule, 242
- invoices
 - Pro Forma, 143
- Invoke a Service field, 311
- Invoke an API field, 311
- Invoke following services as part of this action field, 204
- Invoked Services List field, 204
- Is Active field, 176
- Is Dynamic field, 196
- Is Reprocessible field, 608
- ISO-3166 code, 295
- ISO-639 code, 295
- Issuing Authority field, 117
- Item Classification field, 55, 67
- Item ID field, 64, 65
- i18n tag, 306

J

- jasper printer component, 628
 - auditing, 628
 - configuration properties, 629
 - variable file names, 629
- Java Class field, 185, 247
- Java Server Page field, 307, 308, 310, 313
- JavaScript field, 314
- JMS Queue Name field, 167
- JMS Security, 169, 596, 602
- JMS security, 590

- JMS security properties, 170
 - WebLogic, 170
 - WebSphere, 171
- jsp list, 305
- JSP tag library
 - i18n, 306

L

- Label Format field, 44, 47, 70
- Label Format File Name, 383
- Label Format Name, 383
- language
 - locale, 295
- language codes
 - creating, 280
 - deleting, 281
 - modifying, 280
- Language field, 295
- legal entity, 35
- Legal Entity field, 35
- less-than truckload. See LTL
- License Expiration Date field, 84
- load base document type, 132
- load execution process type, 133
- locale
 - country, 295
 - ISO-3166 code, 295
 - ISO-639 code, 295
 - language, 295
- Locale Code field, 232, 295
- Locale Description field, 295
- Locale field, 32, 34
- locales, 293
 - creating, 294
 - deleting, 296
 - modifying, 296
- Loftware Label Manager, 380
- Log Audits For Draft Order field, 141
- Log File Name field, 244
- LTL carrier preferences
 - nodes, 96
- LTL (Less-than Truck Load), 76

M

Maintain Inventory Cost field, 85
Maintained Externally field, 53
Maintained in Yantra field, 53
Mandate tag information for all tag controlled items field, 63
manifesting process type, 133
MANIFESTPRG purge rule, 242
Mapping Type field, 343
Mapping XML File Name, 383
Mark For field, 50, 64, 65
marketplaces, 3, 37
Maximum Length Of Ship Advice Number field, 254
menu configuration, 297
 defining menu items, 299
 modifying application menu details, 298
 saving a menu group as a new menu grouping, 297
Menu group, 231
Menu Group field, 231
Menu groups
 saving as a new grouping, 297
menu groups, 297
Menu ID field, 300
menu items
 creating, 299
 deleting, 301
 modifying, 301
menu sequence, 301
Menu Sequence field, 301
monitor rule types, 155
Monitoring rules
 defining, 153
Month Display Date Format field, 296
Move Alert To Different Queue field, 348, 349
move request execution process type, 133
multi-divisional corporations, 2, 37

N

Name field, 54, 114
No. of Threads field, 167
Node field, 60
node properties

 configuring, 214
Node Type field, 84
nodes, 38
 defining attributes, 83
 defining LTL carrier preferences, 96
 defining node operations, 86
 defining parcel carrier preferences, 90
 defining primary information, 83
 defining PRO Number, 96
 defining roles
 defining node advanced attributes, 104
 roles
 nodes, 38
nomenclature, 6
nomenclature configuration
 defining, 340
nomenclature definitions
 creating a mapping between, 338
Nomenclature field, 343
nomenclature runtime components, 335
 nomenclature definitions, 338
Nomenclature runtime nodes, 619
nomenclature transformation engine, 335
Non-working day, 124
Not Maintained field, 52
Notification Service field, 99
Notification Threshold field, 99

O

order base document type, 132
Order Creation Tab field, 141
order fulfillment process type, 132
order holds
 deprecated functionality, 253
order negotiation process type, 132
order pipeline, 253
Order # field, 50
Organization Code field, 32, 337
Organization Defines Its Own Catalog field, 111
Organization Is An Enterprise field, 33
organization levels, 12
 rules, 13
Organization Name field, 32
organization rules, 13

- loading another organization's rules, 17
 - overriding, 14
 - organizational hierarchies, 127
 - adding organizations to, 128
 - creating, 127
 - removing organizations from, 128
 - organizations
 - assigning participant associations, 36
 - assigning roles, 36
 - communication protocols, 113
 - creating, 113
 - deleting, 115
 - modifying, 114
 - communication standards
 - communication protocols, 113
 - creating, 31
 - defining calendars, 118
 - creating an exception, 123
 - defining defaults, 122
 - defining payment information, 115
 - defining primary information, 33
 - defining roles
 - defining Carrier attributes, 73
 - defining carrier attributes
 - defining less-than truckload carrier services, 76
 - defining truckload carrier services, 73
 - setting up parcel carrier services, 79
 - defining enterprise attributes, 40
 - adding associated organizations, 41
 - modifying associated organizations, 42
 - removing associated organizations, 43
 - defining node attributes, 83
 - defining LTL carrier preferences, 96
 - defining node operations, 86
 - defining parcel carrier preferences, 90
 - defining primary information, 83
 - defining seller attributes, 45, 48
 - departments
 - creating, 125
 - deleting, 126
 - modifying, 126
 - modifying, 127
 - setting up roles
 - setting up enterprise attributes
 - setting up carrier preferences, 44, 47
 - viewing child organizations, 117
 - viewing departments, 125
 - Other Currency field, 291
 - Other tab, 163
 - outbound picking process type, 133
 - outbound shipment process type, 132
 - Output Name Space field, 307, 308, 310
 - over pack build process type, 133
 - Override Entity ID field, 313
 - Override Freight Terms field, 60
 - Override Modification Rules field, 225
 - Override Ship To field, 60
- ## P
-
- pack process process type, 133
 - Package Level Integration field, 75, 78, 82
 - packaging parameters, 62
 - packing slip, 377
 - Parameter Name, 373
 - Parameter Name field, 169
 - Parameter Value, 373
 - Parameter Value field, 170
 - parcel carrier preferences
 - nodes, 90
 - parcels, 79
 - Parent Organization field, 36
 - Participant modeling, 31
 - participant modeling, 4
 - creating an organization, 31
 - participants, 31
 - pass-through nodes, 211
 - Password field, 231
 - payment information
 - defining an organization's, 115
 - permissions, 223
 - list of users, 224
 - pickup statuses, 189
 - adding, 178
 - deleting, 179
 - pickup Statuses field, 164
 - pipeline determination, 148

- pipelines
 - creating, 149
 - defining monitoring rules, 153
 - modifying, 152
- planned order execution process type, 132
- planned order negotiation process type, 132
- Postal Codes Define This Region field, 361
- presentation, 5
- Primary Contact Address field, 36
- Primary Enterprise field, 33
- primary enterprises, 39
- Primary URL field, 34
- Print Document field, 44, 46, 70
- print documents, 379
 - creating, 378
 - defining, 378
 - deleting, 380
 - modifying, 380
- print format preferences
 - defining, 43, 45, 69
- Printer ID, 229, 236
- printers, 366
- prints
 - defining, 377
 - label formats
 - creating, 381
 - defining, 381
 - deleting, 384
 - modifying, 383
 - Loftware
 - label formats, 381
 - mapping XML file, 381
 - Loftware Label Manager, 381
 - printer - station association, 227, 234
 - printer preferences
 - users
 - setting up, 234
 - user printer preferences
 - defining, 227, 233
- Prior Errors User Exit field, 608
- Priority field, 59
- Pro Forma invoices, 143
- PRO Number End field, 98, 99
- Pro Number Generation Scheme field, 98
- PRO Number Prefix field, 98, 99
- PRO Number Set - 1 field, 98
- PRO Number Set - 2 field, 99
- PRO Number Sets field, 98
- PRO Number Start field, 98, 99
- process modeling, 4, 131
- process modeling tree
 - viewing, 134
- Process Type field, 139
- Process Type Name field, 139
- process type pipelines, 146
- process types, 4, 132, 137
 - count execution, 133
 - defining primary information, 138
 - defining templates, 145
 - general, 133
 - inbound shipment, 133
 - load execution, 133
 - manifesting, 133
 - move request execution, 133
 - order fulfillment, 132
 - order negotiation, 132
 - outbound picking, 133
 - outbound shipment, 132
 - over pack build, 133
 - planned order execution, 132
 - planned order negotiation, 132
 - purchase order execution, 133
 - purchase order negotiation, 133
 - purchase order receipt, 133
 - return receipt, 133
 - return shipment, 133
 - reverse logistics, 132
 - task execution, 133
 - template order, 133
 - trailer loading, 133
 - transfer order delivery, 133
 - transfer order execution, 133
 - transfer order receipt, 133
 - VAS process, 133
 - WMS inventory, 133
 - WMS layout definition, 133
 - WMS putaway, 133
- process tytypespe
 - pack process, 133
- Procure To Ship Allowed field, 88

- procurement, 88
 - chained orders, 88
 - purchase orders, 88
 - transfer orders, 88
- procurement purchase orders, 144
- procurement transfer orders, 144
- Product Class field, 64, 65
- Program field, 205
- protocol codes, 327
 - creating, 328
 - deleting, 328
 - modifying, 328
- Protocol field, 114
- Provider URL field, 169
- Publish Data field, 206
- Publish Date field, 54
- purchase order execution process type, 133
- purchase order negotiation process type, 133
- purchase order receipt process type, 133
- Purge Code field, 243
- purge criteria, 241
 - modifying the system's, 242
- purges, 241

Q

- QCF Lookup field, 168
- Queue Description field, 346
- Queue ID field, 346
- Queue management, 345
- queue management, 6, 363, 377
- Queue Name field, 204, 346, 605
- Queue Priority field, 347
- queues, 345
 - based on size, 347
 - creating, 345
 - deleting, 350
 - modifying, 350
 - unassigned alerts, 348
 - unresolved alerts, 349
 - user list, 350

R

- Raise Action field, 348, 349

- Re-Alert After field, 348, 349
- Receipt Processing Time (Hours) field, 88
- receiver, 209
- Receiving Calendar field, 89
- Redirector View field, 309
- region definition, 6
- region definitions, 353
 - region levels, 353
 - region schemas, 357
 - regions, 360
- Region Level Name field, 354
- region levels
 - creating, 353
 - deleting, 356
 - modifying, 355
- Region Name field, 360
- Region Schema For Routing field, 52
- Region Schema Name field, 359
- region schemas
 - creating, 357
 - deleting, 362
 - modifying, 361
- regions
 - creating, 360, 361
 - deleting, 361
- related entity, 307
- Related Entity field, 307
- repositories, 147
- Represents a Return field, 140
- REPROCESSPRG purge rule, 242
- Requested Delivery Date field, 140
- Requested Ship Date field, 140
- Requires Backward Compatibility field, 176, 185, 247
- Requires Serial Number Tracking field, 85
- Resource ID, 301
- Resource ID field, 301, 303
- Resource Sequence field, 304
- resource sequencing, 304
- Resource Type field, 304
- resources, 302
 - console actions, 314
 - console APIs, 310
 - console detail views
 - console detail views, 308

- console entities, 305
 - viewing jsp list, 305
 - viewing user list, 305
- console icons, 317
- console inner panels, 313
- console links, 316
- console list views, 309
- console related entities, 306
- console search view, 307
- Restrict Access To A Specific List Of Enterprises field, 239
- Restrict List To A Specific List Of Document Types field, 239
- Retention Days field, 243
- return receipt process type, 133
- return shipment process type, 133
- reverse logistics process type, 132
- RF Scanners, 366
- roles
 - assigning to an organization, 39
 - buyers, 38
 - multiple organizations, 37
 - organization, 36
 - sellers, 38
- Roles & Participation tab, 39
- Rollback Segment field, 243
- router nodes, 620
- routing guide lines, 56
- Routing Guide # field, 54
- routing guides, 51
 - fields, 52
 - VICS, 52
- Run Quantity field, 68
- Runtime ID field, 605
- Runtime tab, 605

S

- Save Directory, 380
- SCM not required field, 64
- SCM required with content field, 64
- SCM required without content field, 64
- security, 4, 221
- security management, 221
- Select Algorithm field, 100

- Selection Key Name field, 315
- seller documents, 332
- sellers, 3, 38
 - defining attributes, 45
 - print format preferences, 45
- Send E-Mail field, 205
- Send Fax field, 206
- Send to Alert Console field, 204
- sender, 209
- Sequence No field, 72
- Server Name field, 606
- Server tab, 606
- service definitions, 208, 245
 - creating, 211
- Service Description field, 74, 77, 81
- Service Group Name field, 204, 213
- Service Name field, 185, 204, 213, 247, 311
- service nodes, 208
- Set up data for System Generated scheme field, 98
- Shift Name field, 120, 122
- Shift Valid For field, 123
- shifts, 89, 120
- Ship By Air field, 75, 78, 82
- Shipment Level Integration field, 75, 78, 82
- Shipping Calendar field, 89
- shipping labels, 377
- Skip Automatic Execution field, 312
- Spawning process type field, 163
- Spawns another process? field, 163
- start nodes, 211
- Start Time field, 120, 122
- State field, 58
- STATISTICSPRG purge rule, 242
- status change listeners
 - creating, 186
 - transactions, 186
- status monitoring rule, 192
- status monitoring rule definitions
 - defining, 192
- statuses, 189, 253
- Stored Procedure field, 206
- Store# field, 59, 60
- Subscribed field, 233
- Substitution Is Allowed field, 88
- Supercedes Routing Guide # field, 54

Suspend API field, 607
Suspend Wait Time field, 607
synchronous MQSeries nodes, 589
synchronous transport nodes, 209
synchronous WebLogic nodes, 589
System Generated field, 98
System ID field, 206
System Name field, 337

T

task execution process type, 133
Tax Exempt field, 116
Tax Jurisdiction field, 117
Tax Payer ID field, 116
Template Document Type field, 141
Template field, 205, 206
template order process type, 133
templates, 145
text translator components
 schema files, 636
 XSD files, 635
 delimited receiver, 648
 delimited sender, 654
 positional receiver, 641
 positional sender, 651
text translator file formats
 delimited, 634
 positional, 633
 XML, 634
text translator nodes, 621
Theme field, 232, 319
themes, 318
 creating, 318
 deleting, 319
 modifying, 319
Third Party Logistics Node field, 85
third-party logistics, 37
 nodes, 85
third-party logistics models, 3
This Organization Is A Capacity Organization
 field, 112
This Organization Is An Inventory Organization
 field, 108
This Region Level Can Be Defined By A Set Of

 Postal Codes field, 355
This Service Provides Real-Time Response
 field, 213
This Transaction Can Be Stopped From Processing
 Orders That Are On Hold field, 164
This Transaction Is Derived From Abstract
 field, 163
This transaction is task based field, 163
Time Format field, 296
time formats
 creating, 283
 deleting, 284
 modifying, 284
time unit of measure
 creating, 274
 creating conversion rate, 274
 deleting, 275
 deleting conversion rate, 275
 modifying, 274
 modifying conversion rate, 275
Time Zone field, 295
time-triggered transactions, 159
 defining, 166
To field, 59
To Node field, 94
To Organization field, 341
To System field, 341
trailer loading process type, 133
Transaction ID field, 162
Transaction Name field, 162
transactions, 131, 158
 adding events, 175
 creating, 160
 creating a derived transaction, 185
 creating a status change listener, 186
 defining as externally-triggered, 164
 defining as user-triggered, 174
 defining event handlers, 177
 deleting, 189
 deleting an event, 177
 drop statuses
 adding, 180
 deleting, 181
 externally-triggered, 159
 managing user exits, 181

- modifying, 188
- modifying an event, 176
- pickup statuses
 - adding, 178
 - deleting, 179
- time-triggered, 159
 - defining, 166
- user-triggered, 159
- transfer order delivery process type, 133
- transfer order execution process type, 133
- transfer order receipt process type, 133
- transfer orders, 92
- transfer schedules, 93
- transport nodes, 208, 559
- truckloads, 73
- tThird-party logistics models, 2

U

- UCC-128, 377
- unit, 275
- Unit Of Measure field, 64, 65
- Units of Measure. See UOM
- UOMs (Units of Measure)
 - defining, 261, 264, 266, 269, 271, 273
 - dimension, 266
 - modifying, 262
 - Quantity, 261
 - service quantity, 264
 - volume, 269
- UPS Standard carrier labels, 377
- URL field, 206
- Use as a Template Order field, 140
- Use Default Values field, 343
- Use End Of Shift Time field, 89
- Use Numeric Ship Advice Number flag, 253
- Use Template Order For Defaulting field, 141
- Use units of measure from buyer catalog field, 63
- user, 230
 - creating and modifying, 230
 - deleting, 236
- User Exit Implementation Notes field, 185, 248
- user exit implementations
 - defining, 245
- user exits, 182, 244

- user groups, 221
 - administering resource permissions, 223
 - creating and modifying, 222
 - deleting, 230
 - viewing subscribed users, 226
- User ID field, 204, 231
- user interface extensibility, 5
- user list, 305
- User Name field, 231
- user-triggered transactions, 159
 - defining, 174

V

- Validate Item During Order Creation/ Modification field, 140
- Value field, 114
- VAS process type, 133
- Version field, 185, 248
- VICS, 52
- VICS Bill Of Lading (BOL), 377
- View Group ID field, 308, 309, 310, 315
- View ID field, 315
- View Values field, 337
- volume unit of measure
 - creating, 269
 - creating conversion rate, 270
 - deleting, 271
 - deleting conversion rate, 270
 - modifying, 269
 - modifying conversion rate, 270
- Volume UOM field, 295

W

- wave release prints document set, 377
- WebServices nodes, 587
- weight, 271
- weight unit of measure
 - creating, 271
 - creating conversion rate, 272
 - deleting, 273
 - deleting conversion rate, 273
 - modifying, 272
 - modifying conversion rate, 272

Weight UOM field, 295
When Agent/Integration Server Terminates
Unexpectedly field, 259
When Application Server Goes Down field, 259
When Monitoring Threshold Reached or Exceeded
field, 259
WMS inventory process type, 133
WMS layout definition process type, 133
WMS putaway process type, 133
Work Order Creation Is Allowed field, 88
workflows, 131
working days, 124
Works based on data from field, 163
Write to Log File field, 244

X

XML binding, 315, 317
XML Name field, 341, 342
XML templates, 145
XSL translator nodes, 623

Y

Yantra Application Consoles field, 85
Yantra Networked WMS field, 85
Yantra WMS, 85
Yantra WMS 6.2 field, 86

Z

Zebra 170, 366
Zip Code field, 58

