

Selling and Fulfillment Foundation: Application Platform Configuration Guide

Release 8.5

October 2009



Copyright Notice

Copyright © 1999 - 2009

Sterling Commerce, Inc.

ALL RIGHTS RESERVED

STERLING COMMERCE SOFTWARE

TRADE SECRET NOTICE

THE STERLING COMMERCE SOFTWARE DESCRIBED BY THIS DOCUMENTATION ("STERLING COMMERCE SOFTWARE") IS THE CONFIDENTIAL AND TRADE SECRET PROPERTY OF STERLING COMMERCE, INC., ITS AFFILIATED COMPANIES OR ITS OR THEIR LICENSORS, AND IS PROVIDED UNDER THE TERMS OF A LICENSE AGREEMENT. NO DUPLICATION OR DISCLOSURE WITHOUT PRIOR WRITTEN PERMISSION. RESTRICTED RIGHTS.

This documentation, the Sterling Commerce Software it describes, and the information and know-how they contain constitute the proprietary, confidential and valuable trade secret information of Sterling Commerce, Inc., 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 Sterling Commerce entity. This documentation and the Sterling Commerce Software that 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 Sterling Commerce, Inc. copyright notice. Commerce, Inc. copyright notice.

U.S. GOVERNMENT RESTRICTED RIGHTS. This documentation and the Sterling Commerce Software it describes are "commercial items" as defined in 48 C.F.R. 2.101. 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 Sterling Commerce commercial license agreement are imposed on Government Licensees per 48 C.F.R. 12.212 or § 227.7202 through § 227.7202-4, as applicable, or through 48 C.F.R. § 52.244-6.

This Trade Secret Notice, including the terms of use herein is governed by the laws of the State of Ohio, USA, without regard to its conflict of laws provisions. If you are accessing the Sterling Commerce Software under an executed agreement, then nothing in these terms and conditions supersedes or modifies the executed agreement.

Sterling Commerce, Inc.
4600 Lakehurst Court
Dublin, Ohio 43016-2000

Copyright © 1999 - 2009

Third-Party Software

Portions of the Sterling Commerce Software may include products, or may be distributed on the same storage media with products, ("Third Party Software") offered by third parties ("Third Party Licensors"). Sterling Commerce Software may include Third Party Software covered by the following copyrights: Copyright © 2006-2008 Andres Almiray. Copyright © 1999-2005 The Apache Software Foundation. Copyright (c) 2008 Azer Koçulu <http://azer.kodfabrik.com>. Copyright © Einar Lielmanis, einars@gmail.com. Copyright (c) 2006 John Reilly (www.inconspicuous.org) and Copyright (c) 2002 Douglas Crockford (www.crockford.com). Copyright (c) 2009 John Resig, <http://jquery.com/>. Copyright © 2006-2008 Json-lib. Copyright © 2001 LOOX Software, Inc. Copyright © 2003-2008 Luck Consulting Pty. Ltd. Copyright 2002-2004 © MetaStuff, Ltd. Copyright © 2009 Michael Mathews micmath@gmail.com. Copyright © 1999-2005 Northwoods Software Corporation. Copyright (C) Microsoft Corp. 1981-1998. Purple Technology, Inc. Copyright (c) 2004-2008 QOS.ch. Copyright © 2005 Sabre Airline Solutions. Copyright © 2004 SoftComplex, Inc. Copyright © 2000-2007 Sun Microsystems, Inc. Copyright © 2001 VisualSoft Technologies Limited. Copyright © 2001 Zero G Software, Inc. All rights reserved by all listed parties.

The Sterling Commerce Software is distributed on the same storage media as certain Third Party Software covered by the following copyrights: Copyright © 1999-2006 The Apache Software Foundation. Copyright (c) 2001-2003 Ant-Contrib project. Copyright © 1998-2007 Bela Ban. Copyright © 2005 Eclipse Foundation. Copyright © 2002-2006 Julian Hyde and others. Copyright © 1997 ICE Engineering, Inc./Timothy Gerard Endres. Copyright 2000, 2006 IBM Corporation and others. Copyright © 1987-2006 ILOG, Inc. Copyright © 2000-2006 Infragistics. Copyright © 2002-2005 JBoss, Inc. Copyright LuMriX.net GmbH, Switzerland. Copyright © 1998-2009 Mozilla.org. Copyright © 2003-2009 Mozdev Group, Inc. Copyright © 1999-2002 JBoss.org. Copyright Raghu K, 2003. Copyright © 2004 David Schweinsberg. Copyright © 2005-2006 Darren L. Spurgeon. Copyright © S.E. Morris (FISH) 2003-04. Copyright © 2006 VisualSoft Technologies. Copyright © 2002-2009 Zipwise Software. All rights reserved by all listed parties.

Certain components of the Sterling Commerce Software are distributed on the same storage media as Third Party Software which are not listed above. Additional information for such Third Party Software components of the Sterling Commerce Software is located at: `installdir/ mesa/studio/plugins/SCI_Studio_License.txt`.

Third Party Software which is included, or are distributed on the same storage media with, the Sterling Commerce Software where use, duplication, or disclosure by the United States government or a government contractor or subcontractor, are provided with RESTRICTED RIGHTS under Title 48 CFR 2.101, 12.212, 52.227-19, 227.7201 through 227.7202-4, DFAR 252.227-7013(c) (1) (ii) and (2), DFAR 252.227-7015(b)(6/95), DFAR 227.7202-3(a), FAR 52.227-14(g)(2)(6/87), and FAR 52.227-19(c)(2) and (6/87) as applicable.

Additional information regarding certain Third Party Software is located at `installdir/SCI_License.txt`.

Some Third Party Licensors also provide license information and/or source code for their software via their respective links set forth below:

<http://danadler.com/jacob/>

<http://www.dom4j.org>

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>). This product includes software developed by the Ant-Contrib project (<http://sourceforge.net/projects/ant-contrib>). This product includes software developed by the JDOM Project (<http://www.jdom.org/>). This product includes code licensed from RSA Data Security (via Sun Microsystems, Inc.). Sun, Sun Microsystems, the Sun Logo, Java, JDK, the Java Coffee Cup logo, JavaBeans, JDBC, JMX and all JMX based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. All other trademarks and logos are trademarks of their respective owners.

THE APACHE SOFTWARE FOUNDATION SOFTWARE

The Sterling Commerce Software is distributed with or on the same storage media as the following software products (or components thereof) and java source code files: Xalan version 2.5.2, Cookie.java, Header.java, HeaderElement.java, HttpException.java, HttpState.java, NameValuePair.java, CronTimeTrigger.java, DefaultTimeScheduler.java, PeriodicTimeTrigger.java, Target.java,

TimeScheduledEntry.java, TimeScheduler.java, TimeTrigger.java, Trigger.java, BinaryHeap.java, PriorityQueue.java, SynchronizedPriorityQueue.java, GetOpt.java, GetOptsException.java, IllegalArgumentException.java, MissingOptArgException.java (collectively, "Apache 1.1 Software"). Apache 1.1 Software is free software which is distributed under the terms of the following license:

License Version 1.1

Copyright 1999-2003 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement: "This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).". Alternatively, this acknowledgement may appear in the software itself, if and whenever such third-party acknowledgements normally appear.
4. The names "Commons", "Jakarta", "The Jakarta Project", "HttpClient", "log4j", "Xerces", "Xalan", "Avalon", "Apache Avalon", "Avalon Cornerstone", "Avalon Framework", "Apache" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without specific prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without the prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. The GetOpt.java, GetOptsException.java, IllegalArgumentException.java and MissingOptArgException.java software was originally based on software copyright (c) 2001, Sun Microsystems., <http://www.sun.com>. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

The preceding license only applies to the Apache 1.1 Software and does not apply to the Sterling Commerce Software or to any other Third-Party Software.

The Sterling Commerce Software is also distributed with or on the same storage media as the following software products (or components thereof): Ant, Antinstaller, Apache File Upload Package, Apache Commons Beans, Apache Commons BetWixt, Apache Commons Collection, Apache Commons Digester, Apache Commons IO, Apache Commons Lang., Apache Commons Logging, Apache Commons Net, Apache Jakarta Commons Pool, Apache Jakarta ORO, Lucene, Xerces version 2.7, Apache Log4J, Apache SOAP, Apache Struts and Apache Xalan 2.7.0. (collectively, "Apache 2.0 Software"). Apache 2.0 Software is free software which is distributed under the terms of the Apache License Version 2.0. A copy of License Version 2.0 is found in the following directory files for the individual pieces of the Apache 2.0 Software: `install/jar/commons_upload/1_0/ CommonsFileUpload_License.txt`, `install/jar/jetspeed/1_4/RegExp_License.txt`, `install/ant/Ant_License.txt`, `<install>/jar/antInstaller/0_8/antinstaller_License.txt`, `<install>/jar/commons_beanutils/1_7_0/commons-beanutils.jar (/META-INF/LICENSE.txt)`, `<install>/jar/commons_betwixt/0_8/commons-betwixt-0.8.jar (/META-INF/LICENSE.txt)`,

```

<install>/jar/commons_collections/3_2/LICENSE.txt,
<install>/jar/commons_digester/1_8/commons-digester-1.8.jar (/META-INF/LICENSE.txt),
<install>/jar/commons_io/1_4/LICENSE.txt,
<install>/jar/commons_lang/2_1/Commons_Lang_License.txt,
<install>/jar/commons_logging/1_0_4/commons-logging-1.0.4.jar (/META-INF/LICENSE.txt),
<install>/jar/commons_net/1_4_1/commons-net-1.4.1.jar (/META-INF/LICENSE.txt),
<install>/jar/smcfs/8.5/lucene-core-2.4.0.jar (/META-INF/LICENSE.txt),
<install>/jar/struts/2_0_11/struts2-core-2.0.11.jar (./LICENSE.txt),
<install>/jar/mesa/gisdav/WEB-INF/lib/Slide_License.txt,
<install>/mesa/studio/plugins/xerces_2_7_license.txt,
<install>/jar/commons_pool/1_2/Commons_License.txt,
<install>/jar/jakarta_oro/2_0_8/JakartaOro_License.txt,
<install>/jar/log4j/1_2_15/LOG4J_License.txt,
<install>/jar/xalan/2_7/Xalan_License.txt,
<install>/jar/soap/2_3_1/Apache_SOAP_License.txt

```

Unless otherwise stated in a specific directory, the Apache 2.0 Software was not modified. Neither the Sterling Commerce Software, modifications, if any, to Apache 2.0 Software, nor other Third Party Code is a Derivative Work or a Contribution as defined in License Version 2.0. License Version 2.0 applies only to the Apache 2.0 Software which is the subject of the specific directory file and does not apply to the Sterling Commerce Software or to any other Third Party Software. License Version 2.0 includes the following provision:

"Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License."

NOTICE file corresponding to the section 4 d of the Apache License, Version 2.0, in this case for the Apache Ant distribution. Apache Ant Copyright 1999-2008 The Apache Software Foundation. This product includes software developed by The Apache Software Foundation (<http://www.apache.org/>). This product includes also software developed by :

- the W3C consortium (<http://www.w3c.org>) ,
- the SAX project (<http://www.saxproject.org>)

The <sync> task is based on code Copyright (c) 2002, Landmark Graphics Corp that has been kindly donated to the Apache Software Foundation.

Portions of this software were originally based on the following:

- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

NOTICE file corresponding to the section 4 d of the Apache License, Version 2.0, in this case for the Apache Lucene distribution. Apache Lucene Copyright 2006 The Apache Software Foundation. This product includes software developed by The Apache Software Foundation (<http://www.apache.org/>). The snowball stemmers in contrib/snowball/src/java/net/sf/snowball were developed by Martin Porter and Richard Boulton. The full snowball package is available from <http://snowball.tartarus.org/>

Ant-Contrib Software

The Sterling Commerce Software is distributed with or on the same storage media as the Anti-Contrib software (Copyright (c) 2001-2003 Ant-Contrib project. All rights reserved.) (the "Ant-Contrib Software"). The Ant-Contrib Software is free software which is distributed under the terms of the following license:

The Apache Software License, Version 1.1

Copyright (c) 2001-2003 Ant-Contrib project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement:

"This product includes software developed by the Ant-Contrib project (<http://sourceforge.net/projects/ant-contrib>)."

Alternately, this acknowledgement may appear in the software itself, if and wherever such third-party acknowledgements normally appear.

4. The name Ant-Contrib must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact ant-contrib-developers@lists.sourceforge.net.

5. Products derived from this software may not be called "Ant-Contrib" nor may "Ant-Contrib" appear in their names without prior written permission of the Ant-Contrib project.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE ANT-CONTRIB PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. The preceding license only applies to the Ant-Contrib Software and does not apply to the Sterling Commerce Software or to any other Third-Party Software.

The preceding license only applies to the Ant-Contrib Software and does not apply to the Sterling Commerce Software or to any other Third Party Software.

DOM4J Software

The Sterling Commerce Software is distributed with or on the same storage media as the Dom4h Software which is free software distributed under the terms of the following license:

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name "DOM4J" must not be used to endorse or promote products derived from this Software without prior written permission of MetaStuff, Ltd. For written permission, please contact dom4j-info@metastuff.com.
4. Products derived from this Software may not be called "DOM4J" nor may "DOM4J" appear in their names without prior written permission of MetaStuff, Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.
5. Due credit should be given to the DOM4J Project - <http://www.dom4j.org>

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2001-2004 (C) MetaStuff, Ltd. All Rights Reserved.

The preceding license only applies to the Dom4j Software and does not apply to the Sterling Commerce Software, or any other Third-Party Software.

THE ECLIPSE SOFTWARE FOUNDATION

The Sterling Commerce Software is also distributed with or on the same storage media as the following software:

com.ibm.icu.nl1_3.4.4.v200606220026.jar, org.eclipse.ant.core.nl1_3.1.100.v200606220026.jar,
org.eclipse.ant.ui.nl1_3.2.0.v200606220026.jar, org.eclipse.compare.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.boot.nl1_3.1.100.v200606220026.jar,
org.eclipse.core.commands.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.contenttype.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.expressions.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.filebuffers.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.filesystem.nl1_1.0.0.v200606220026.jar,
org.eclipse.core.jobs.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.resources.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.runtime.compatibility.auth.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.runtime.compatibility.nl1_3.1.100.v200606220026.jar,
org.eclipse.core.runtime.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.variables.nl1_3.1.100.v200606220026.jar,
org.eclipse.debug.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.debug.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.equinox.common.nl1_3.2.0.v200606220026.jar,
org.eclipse.equinox.preferences.nl1_3.2.0.v200606220026.jar,
org.eclipse.equinox.registry.nl1_3.2.0.v200606220026.jar,
org.eclipse.help.appserver.nl1_3.1.100.v200606220026.jar,
org.eclipse.help.base.nl1_3.2.0.v200606220026.jar, org.eclipse.help.nl1_3.2.0.v200606220026.jar,
org.eclipse.help.ui.nl1_3.2.0.v200606220026.jar, org.eclipse.jdt.apt.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.apt.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.core.manipulation.nl1_1.0.0.v200606220026.jar,
org.eclipse.jdt.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.debug.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.doc.isv.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.doc.user.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.junit4.runtime.nl1_1.0.0.v200606220026.jar,
org.eclipse.jdt.launching.nl1_3.2.0.v200606220026.jar, org.eclipse.jdt.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.jface.databinding.nl1_1.0.0.v200606220026.jar,
org.eclipse.jface.nl1_3.2.0.v200606220026.jar, org.eclipse.jface.text.nl1_3.2.0.v200606220026.jar,
org.eclipse.ltk.core.refactoring.nl1_3.2.0.v200606220026.jar,
org.eclipse.ltk.ui.refactoring.nl1_3.2.0.v200606220026.jar,
org.eclipse.osgi.nl1_3.2.0.v200606220026.jar, org.eclipse.osgi.services.nl1_3.1.100.v200606220026.jar,
org.eclipse.osgi.util.nl1_3.1.100.v200606220026.jar, org.eclipse.pde.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.pde.doc.user.nl1_3.2.0.v200606220026.jar,
org.eclipse.pde.junit.runtime.nl1_3.2.0.v200606220026.jar,
org.eclipse.pde.nl1_3.2.0.v200606220026.jar, org.eclipse.pde.runtime.nl1_3.2.0.v200606220026.jar,
org.eclipse.pde.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.platform.doc.isv.nl1_3.2.0.v200606220026.jar,
org.eclipse.platform.doc.user.nl1_3.2.0.v200606220026.jar,

org.eclipse.rcp.nl1_3.2.0.v200606220026.jar, org.eclipse.search.nl1_3.2.0.v200606220026.jar,
org.eclipse.swt.nl1_3.2.0.v200606220026.jar, org.eclipse.team.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.team.cvs.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.team.cvs.ssh.nl1_3.2.0.v200606220026.jar,
org.eclipse.team.cvs.ssh2.nl1_3.2.0.v200606220026.jar,
org.eclipse.team.cvs.ui.nl1_3.2.0.v200606220026.jar, org.eclipse.team.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.text.nl1_3.2.0.v200606220026.jar, org.eclipse.ui.browser.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.cheatsheets.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.console.nl1_3.1.100.v200606220026.jar,
org.eclipse.ui.editors.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.externaltools.nl1_3.1.100.v200606220026.jar,
org.eclipse.ui.forms.nl1_3.2.0.v200606220026.jar, org.eclipse.ui.ide.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.intro.nl1_3.2.0.v200606220026.jar, org.eclipse.ui.navigator.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.navigator.resources.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.presentations.r21.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.views.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.views.properties.tabbed.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.workbench.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.workbench.texteditor.nl1_3.2.0.v200606220026.jar,
org.eclipse.update.configurator.nl1_3.2.0.v200606220026.jar,
org.eclipse.update.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.update.scheduler.nl1_3.2.0.v200606220026.jar,
org.eclipse.update.ui.nl1_3.2.0.v200606220026.jar,
com.ibm.icu_3.4.4.1.jar,
org.eclipse.core.commands_3.2.0.I20060605-1400.jar,
org.eclipse.core.contenttype_3.2.0.v20060603.jar,
org.eclipse.core.expressions_3.2.0.v20060605-1400.jar,
org.eclipse.core.filesystem.linux.x86_1.0.0.v20060603.jar,
org.eclipse.core.filesystem_1.0.0.v20060603.jar, org.eclipse.core.jobs_3.2.0.v20060603.jar,
org.eclipse.core.runtime.compatibility.auth_3.2.0.v20060601.jar,
org.eclipse.core.runtime_3.2.0.v20060603.jar, org.eclipse.equinox.common_3.2.0.v20060603.jar,
org.eclipse.equinox.preferences_3.2.0.v20060601.jar, org.eclipse.equinox.registry_3.2.0.v20060601.jar,
org.eclipse.help_3.2.0.v20060602.jar, org.eclipse.jface.text_3.2.0.v20060605-1400.jar,
org.eclipse.jface_3.2.0.I20060605-1400.jar, org.eclipse.osgi_3.2.0.v20060601.jar,
org.eclipse.swt.gtk.linux.x86_3.2.0.v3232m.jar, org.eclipse.swt_3.2.0.v3232o.jar,
org.eclipse.text_3.2.0.v20060605-1400.jar,
org.eclipse.ui.workbench.texteditor_3.2.0.v20060605-1400.jar,
org.eclipse.ui.workbench_3.2.0.I20060605-1400.jar, org.eclipse.ui_3.2.0.I20060605-1400.jar,
runtime_registry_compatibility.jar, eclipse.exe, eclipse.ini, and startup.jar
(collectively, "Eclipse Software").

All Eclipse Software is distributed under the terms and conditions of the Eclipse Foundation Software User Agreement (EFSUA) and/or terms and conditions of the Eclipse Public License Version 1.0 (EPL) or other license agreements, notices or terms and conditions referenced for the individual pieces of the Eclipse Software, including without limitation "Abouts", "Feature Licenses", and "Feature Update Licenses" as defined in the EFSUA .

A copy of the Eclipse Foundation Software User Agreement is found at
<install_dir>/SI/repository/rcp/rcpdependencies/windows/eclipse/notice.html,
<install_dir>/SI/repository/rcp/rcpdependencies/windows/eclipse/plugins/notice.html,
<install_dir>/SI/repository/rcp/rcpdependencies/gtk.linux.x86/eclipse/notice.html, and
<install_dir>/SI/repository/rcp/rcpdependencies/gtk.linux.x86/eclipse/plugins/notice.html.

A copy of the EPL is found at
<install_dir>/SI/repository/rcp/rcpdependencies/windows/eclipse/plugins/epl-v10.htm,
<install_dir>/SI/repository/rcp/rcpdependencies/windows/eclipse/epl-v10.htm,
<install_dir>/SI/repository/rcp/rcpdependencies/gtk.linux.x86/eclipse/plugins/epl-v10.html, and
<install_dir>/SI/repository/rcp/rcpdependencies/gtk.linux.x86/eclipse/epl-v10.html.

The reference to the license agreements, notices or terms and conditions governing each individual piece of the Eclipse Software is found in the directory files for the individual pieces of the Eclipse Software as described in the file identified as installDir/SCI_License.txt.

These licenses only apply to the Eclipse Software and do not apply to the Sterling Commerce Software, or any other Third Party Software.

The Language Pack (NL Pack) piece of the Eclipse Software, is distributed in object code form. Source code is available at http://archive.eclipse.org/eclipse/downloads/drops/L-3.2_Language_Packs-200607121700/index.php. In the event the source code is no longer available from the website referenced above, contact Sterling Commerce at 978-513-6000 and ask for the Release Manager. A copy of this license is located at <install_dir>/SI/repository/rcp/rcpdependencies/windows/eclipse/plugins/epl-v10.htm and <install_dir>/SI/repository/rcp/rcpdependencies/gtk.linux.x86/eclipse/plugins/epl-v10.html.

The org.eclipse.core.runtime_3.2.0.v20060603.jar piece of the Eclipse Software was modified slightly in order to remove classes containing encryption items. The org.eclipse.core.runtime_3.2.0.v20060603.jar was modified to remove the Cipher, CipherInputStream and CipherOutputStream classes and rebuild the org.eclipse.core.runtime_3.2.0.v20060603.jar.

Ehcache Software

The Sterling Commerce Software is also distributed with or on the same storage media as the ehache v.1.5 software (Copyright © 2003-2008 Luck Consulting Pty. Ltd.) ("Ehcache Software"). Ehcache Software is free software which is distributed under the terms of the Apache License Version 2.0. A copy of License Version 2.0 is found in <install>/jar/smcfcs/8.5/ehcache-1.5.0.jar (./LICENSE.txt).

The Ehcache Software was not modified. Neither the Sterling Commerce Software, modifications, if any, to the Ehcache Software, nor other Third Party Code is a Derivative Work or a Contribution as defined in License Version 2.0. License Version 2.0 applies only to the Ehcache Software which is the subject of the specific directory file and does not apply to the Sterling Commerce Software or to any other Third Party Software. License Version 2.0 includes the following provision:

"Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License."

EZMorph Software

The Sterling Commerce Software is also distributed with or on the same storage media as the EZMorph v. 1.0.4 software (Copyright © 2006-2008 Andres Almiray) ("EZMorph Software"). EZMorph Software is free software which is distributed under the terms of the Apache License Version 2.0. A copy of License Version 2.0 is found in <install>/jar/ezmorph/1_0_4/ezmorph-1.0.4.jar (./LICENSE.txt).

The EZMorph Software was not modified. Neither the Sterling Commerce Software, modifications, if any, to the EZMorph Software, nor other Third Party Code is a Derivative Work or a Contribution as defined in License Version 2.0. License Version 2.0 applies only to the EZMorph Software which is the subject of the specific directory file and does not apply to the Sterling Commerce Software or to any other Third Party Software. License Version 2.0 includes the following provision:

"Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License."

Firebug Lite Software

The Sterling Commerce Software is distributed with or on the same storage media as the Firebug Lite Software which is free software distributed under the terms of the following license:

Copyright (c) 2008 Azer Koçulu <http://azer.kodfabrik.com>. All rights reserved.

Redistribution and use of this software in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Azer Koçulu, nor the names of any other contributors may be used to endorse or promote products derived from this software without specific prior written permission of Parakey Inc.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ICE SOFTWARE

The Sterling Commerce Software is distributed on the same storage media as the ICE Software (Copyright © 1997 ICE Engineering, Inc./Timothy Gerard Endres.) ("ICE Software"). The ICE Software is independent from and not linked or compiled with the Sterling Commerce Software. The ICE Software is a free software product which can be distributed and/or modified under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License or any later version.

A copy of the GNU General Public License is provided at `install_dir/jar/jniregistry/1_2/ICE_License.txt`. This license only applies to the ICE Software and does not apply to the Sterling Commerce Software, or any other Third Party Software.

The ICE Software was modified slightly in order to fix a problem discovered by Sterling Commerce involving the RegistryKey class in the RegistryKey.java in the JNIRegistry.jar. The class was modified to comment out the finalize () method and rebuild of the JNIRegistry.jar file.

Source code for the bug fix completed by Sterling Commerce on January 8, 2003 is located at: `install_dir/jar/jniregistry/1_2/RegistryKey.java`. Source code for all other components of the ICE Software is located at <http://www.trustice.com/java/jnireg/index.shtml>.

The ICE Software is distributed WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

JBOSS SOFTWARE

The Sterling Commerce Software is distributed on the same storage media as the JBoss Software (Copyright © 1999-2002 JBoss.org) ("JBoss Software"). The JBoss Software is independent from and not linked or compiled with the Sterling Commerce Software. The JBoss Software is a free software product which can be distributed and/or modified under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License or any later version.

A copy of the GNU Lesser General Public License is provided at:
<install_dir>\jar\jboss\4_2_0\LICENSE.html

This license only applies to the JBoss Software and does not apply to the Sterling Commerce Software, or any other Third Party Software.

The JBoss Software is not distributed by Sterling Commerce in its entirety. Rather, the distribution is limited to the following jar files: `el-api.jar`, `jasper-compiler-5.5.15.jar`, `jasper-el.jar`, `jasper.jar`, `jboss-common-client.jar`, `jboss-j2ee.jar`, `jboss-jmx.jar`, `jboss-jsr77-client.jar`, `jbossmq-client.jar`,

jnpserver.jar, jsp-api.jar, servlet-api.jar, tomcat-juli.jar.

The JBoss Software was modified slightly in order to allow the ClientSocketFactory to return a socket connected to a particular host in order to control the host interfaces, regardless of whether the ClientSocket Factory specified was custom or not. Changes were made to org.jnp.server.Main. Details concerning this change can be found at http://sourceforge.net/tracker/?func=detail&aid=1008902&group_id=22866&atid=376687.

Source code for the modifications completed by Sterling Commerce on August 13, 2004 is located at: http://sourceforge.net/tracker/?func=detail&aid=1008902&group_id=22866&atid=376687. Source code for all other components of the JBoss Software is located at <http://www.jboss.org>.

JGO SOFTWARE

The Sterling Commerce Software is distributed with, or on the same storage media, as certain redistributable portions of the JGo Software provided by Northwoods Software Corporation under a commercial license agreement (the "JGo Software"). The JGo Software is provided subject to the disclaimer set forth above and the following notice:

U.S. Government Restricted Rights

The JGo Software and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (C)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (C)(1) and (2) of the Commercial Computer Software - Restricted Rights at 48 CFR 52.227-19, as applicable. Contractor / manufacturer of the JGo Software is Northwoods Software Corporation, 142 Main St., Nashua, NH 03060.

JSLib Software

The Sterling Commerce Software is distributed with or on the same storage media as the JSLib software product (Copyright (c) 2003-2009 Mozdev Group, Inc.) ("JSLib Software"). The JSLib Software is distributed under the terms of the MOZILLA PUBLIC LICENSE Version 1.1. A copy of this license is located at <install>\repository\ear\data\platform_uifwk_ide\war\designer\MPL-1.1.txt. The JSLib Software code is distributed in source form and is located at <http://jslib.mozdev.org/installation.html>. Neither the Sterling Commerce Software nor any other Third-Party Code is a Modification or Contribution subject to the Mozilla Public License. Pursuant to the terms of the Mozilla Public License, the following notice applies only to the JSLib Software (and not to the Sterling Commerce Software or any other Third-Party Software):

"The contents of the file located at <http://www.mozdev.org/source/browse/jslib/> are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/MPL-1.1.html>.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Mozdev Group, Inc. code. The Initial Developer of the Original Code is Mozdev Group, Inc. Portions created by Mozdev Group, Inc. are Copyright © 2003 Mozdev Group, Inc. All Rights Reserved. Original Author: Pete Collins <pete@mozdev.org> one Contributor(s): _____ none listed _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[_____] License"), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [_____] License and not allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [_____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [_____] License."

The preceding license only applies to the JSLib Software and does not apply to the Sterling Commerce Software, or any other Third-Party Software.

Json Software

The Sterling Commerce Software is also distributed with or on the same storage media as the Json 2.2.2 software (Copyright © 2006-2008 Json-lib) ("Json Software"). Json Software is free software which is distributed under the terms of the Apache License Version 2.0. A copy of License Version 2.0 is found in <install>/jar/jsonlib/2_2_2/json-lib-2.2.2-jdk13.jar.

This product includes software developed by Douglas Crockford (<http://www.crockford.com>).

The Json Software was not modified. Neither the Sterling Commerce Software, modifications, if any, to the Json Software, nor other Third Party Code is a Derivative Work or a Contribution as defined in License Version 2.0. License Version 2.0 applies only to the Json Software which is the subject of the specific directory file and does not apply to the Sterling Commerce Software or to any other Third Party Software. License Version 2.0 includes the following provision:

"Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License."

Purple Technology

The Sterling Commerce Software is distributed with or on the same storage media as the Purple Technology Software (Copyright (c) 1995-1999 Purple Technology, Inc.) ("Purple Technology Software"), which is subject to the following license:

Copyright (c) 1995-1999 Purple Technology, Inc. All rights reserved.

PLAIN LANGUAGE LICENSE: Do whatever you like with this code, free of charge, just give credit where credit is due. If you improve it, please send your improvements to alex@purpletech.com. Check <http://www.purpletech.com/code/> for the latest version and news.

LEGAL LANGUAGE LICENSE: Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors and the names "Purple Technology," "Purple Server" and "Purple Chat" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact server@purpletech.com.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND PURPLE TECHNOLOGY "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR PURPLE TECHNOLOGY BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The preceding license only applies to the Purple Technology Software and does not apply to the Sterling Commerce Software, or any other Third Party Software.

Rico Software

The Sterling Commerce Software is also distributed with or on the same storage media as the Rico.js software (Copyright © 2005 Sabre Airline Solutions) ("Rico Software"). Rico Software is free software

which is distributed under the terms of the Apache License Version 2.0. A copy of License Version 2.0 is found in <install>/repository/eardata/platform/war/ajax/scripts/Rico_License.txt.

The Rico Software was not modified. Neither the Sterling Commerce Software, modifications, if any, to the Rico Software, nor other Third-Party Code is a Derivative Work or a Contribution as defined in License Version 2.0. License Version 2.0 applies only to the Rico Software which is the subject of the specific directory file and does not apply to the Sterling Commerce Software or to any other Third-Party Software. License Version 2.0 includes the following provision:

"Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License."

Rhino Software

The Sterling Commerce Software is distributed with or on the same storage media as the Rhino.js.jar (Copyright (c) 1998-2009 Mozilla.org.) ("Rhino Software"). A majority of the source code for the Rhino Software is dual licensed under the terms of the MOZILLA PUBLIC LICENSE Version 1.1. or the GPL v. 2.0. Additionally, some files (at a minimum the contents of toolsrc/org/Mozilla/javascript/toolsdebugger/treetable) are available under another license as set forth in the directory file for the Rhino Software.

Sterling Commerce's use and distribution of the Rhino Software is under the Mozilla Public License. A copy of this license is located at <install>/3rdParty/rico license.doc. The Rhino Software code is distributed in source form and is located at <http://mxr.mozilla.org/mozilla/source/js/rhino/src/>. Neither the Sterling Commerce Software nor any other Third-Party Code is a Modification or Contribution subject to the Mozilla Public License. Pursuant to the terms of the Mozilla Public License, the following notice applies only to the Rhino Software (and not to the Sterling Commerce Software or any other Third-Party Software):

"The contents of the file located at <install>/jar/rhino/1_7R1/js.jar are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Rhino code, released May 6, 1999. The Initial Developer is Netscape Communications Corporation. Portions created by the Initial Developer are Copyright © 1997-1999. All Rights Reserved. Contributor(s): _____none listed.

The preceding license only applies to the Rico Software and does not apply to the Sterling Commerce Software, or any other Third-Party Software.

Sun Microsystems

The Sterling Commerce Software is distributed with or on the same storage media

as the following software products (or components thereof): Sun JMX, and Sun JavaMail (collectively, "Sun Software"). Sun Software is free software which is distributed under the terms of the licenses issued by Sun which are included in the directory files located at:

SUN COMM JAR - <install>/Applications/Foundation/lib

SUN ACTIVATION JAR - <install>/ Applications/Foundation/lib

SUN JavaMail - <install>/jar/javamail/1_4/LICENSE.txt

The Sterling Commerce Software is also distributed with or on the same storage media as the Web-app_2_3.dtd software (Copyright © 2007 Sun Microsystems, Inc.) ("Web-App Software"). Web-App Software is free software which is distributed under the terms of the Common Development

and Distribution License ("CDDL"). A copy of the CDDL is found in <http://kenai.com/projects/javamail/sources/mercurial/show>.

The source code for the Web-App Software may be found at:
<install>/3rdParty/sun/javamail-1.3.2/docs/JavaMail-1.2.pdf

Such licenses only apply to the Sun product which is the subject of such directory and does not apply to the Sterling Commerce Software or to any other Third Party Software.

The Sterling Commerce Software is also distributed with or on the same storage media as the Sun Microsystems, Inc. Java (TM) look and feel Graphics Repository ("Sun Graphics Artwork"), subject to the following terms and conditions:

Copyright 2000 by Sun Microsystems, Inc. All Rights Reserved.

Sun grants you ("Licensee") a non-exclusive, royalty free, license to use, and redistribute this software graphics artwork, as individual graphics or as a collection, as part of software code or programs that you develop, provided that i) this copyright notice and license accompany the software graphics artwork; and ii) you do not utilize the software graphics artwork in a manner which is disparaging to Sun. Unless enforcement is prohibited by applicable law, you may not modify the graphics, and must use them true to color and unmodified in every way.

This software graphics artwork is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE GRAPHICS ARTWORK.

IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE SOFTWARE GRAPHICS ARTWORK, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

If any of the above provisions are held to be in violation of applicable law, void, or unenforceable in any jurisdiction, then such provisions are waived to the extent necessary for this Disclaimer to be otherwise enforceable in such jurisdiction.

The preceding license only applies to the Sun Graphics Artwork and does not apply to the Sterling Commerce Software, or any other Third Party Software.

WARRANTY DISCLAIMER

This documentation and the Sterling Commerce Software which it describes are licensed either "AS IS" or with a limited warranty, as set forth in the Sterling Commerce 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 Sterling Commerce entity reserves the right to revise this publication from time to time and to make changes in the content hereof without the obligation to notify any person or entity of such revisions or changes.

The Third Party Software is provided "AS IS" WITHOUT ANY WARRANTY AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. FURTHER, IF YOU ARE LOCATED OR ACCESSING THIS SOFTWARE IN THE UNITED STATES, ANY EXPRESS OR IMPLIED WARRANTY REGARDING TITLE OR NON-INFRINGEMENT ARE DISCLAIMED.

Without limiting the foregoing, the ICE Software and JBoss Software are distributed WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

Contents

Preface

Intended Audience	xxxv
Structure	xxxv
Selling and Fulfillment Foundation Documentation	xi
Conventions	xlii

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 Application Platform Configuration	3
1.2.1 Participant Modeling	4
1.2.2 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	6
1.2.9 Nomenclature	6
1.2.10 Queue Management	6
1.2.11 Region Definition	6
1.2.12 Alert Management	6

2 Navigating the Applications Manager

2.1	Starting the Applications Manager	9
2.2	The Applications Manager Layout.....	10
2.2.1	Application Rules Side Panel	12
2.2.1.1	Accessing Configuration Screens.....	13
2.2.1.2	Determining Inheritance.....	14
2.2.1.3	Loading Another Organization's Rules.....	21
2.2.2	Work Area	23
2.2.2.1	Search Window	23
2.2.2.2	List Window.....	24
2.2.2.3	Details Window	25
2.2.2.4	Drag and Drop Window	26
2.3	Actions Available in the Applications Manager.....	28
2.3.1	Using the Applications Manager's Lookup Functionality	28
2.3.2	Viewing the Document Types Associated with an Application	29
2.3.2.1	Adding a Document Type to an Application	30
2.3.3	Viewing the User Logged into the Applications Manager	31
2.3.4	Using Lists and List Filtering.....	31
2.3.5	Date and Time Entry.....	34
2.3.6	Using On-Line Help.....	34
2.3.7	Troubleshooting Errors.....	34
2.3.8	Using Special Characters	35

3 Configuring Participants

3.1	Creating and Modifying an Organization	37
3.1.1	Defining an Organization's Primary Information	42
3.1.2	Assigning the Organization's Roles and Participant Associations	45
3.1.2.1	Defining Enterprise Attributes	49
3.1.2.2	Defining Seller Attributes.....	57
3.1.2.3	Defining Buyer Attributes	62
3.1.2.4	Defining Carrier Attributes	84
3.1.2.5	Defining Node Attributes	96
3.1.2.6	Defining a Node's Advanced Inventory Attributes.....	124
3.1.2.7	Defining an Organization's Advanced Inventory Attributes.....	127

3.1.2.8	Defining an Organization's Advanced Catalog Attributes.....	131
3.1.2.9	Defining an Organization's Advanced Capacity Attributes	133
3.1.2.10	Defining an Organization's Advanced Customer Attributes.....	135
3.1.2.11	Defining an Organization's Advanced Pricing Attributes.....	136
3.1.2.12	Defining an Organization's Themes	137
3.1.3	Assigning a Yard's Roles and Participant Associations	139
3.1.4	Defining Communication Protocols	140
3.1.4.1	Creating a Communication Protocol	140
3.1.4.2	Modifying a Communication Protocol	141
3.1.4.3	Deleting a Communication Protocol.....	142
3.1.4.4	Defining Buyer Documents	142
3.1.4.5	Setting Up Seller Documents	145
3.1.4.6	Setting Up Carrier Documents.....	148
3.1.5	Defining an Organization's Payment Information.....	151
3.1.6	Viewing an Organization's Child Organizations.....	153
3.1.7	Defining an Organization's Calendars	154
3.1.7.1	Defining a Calendar's Defaults.....	158
3.1.7.2	Creating an Exception for a Particular Calendar Day	159
3.1.8	Viewing an Organization's Departments.....	161
3.1.8.1	Creating a Department.....	161
3.1.8.2	Modifying a Department	162
3.1.8.3	Deleting a Department.....	163
3.1.9	Modifying an Organization	163
3.2	Creating and Modifying an Organizational Hierarchy.....	163
3.2.1	Creating an Organizational Hierarchy	164
3.2.1.1	Adding an Organization to the Organizational Hierarchy.....	164
3.2.1.2	Removing an Organization from the Organizational Hierarchy.....	165
3.3	Creating Node Types	165
3.3.1	Creating a Node Type	165
3.3.2	Modifying a Node Type.....	166
3.3.3	Deleting a Node Type	166

4 Configuring Process Models

4.1	Document Type Configuration	167
4.1.1	The Process Modeling Tree	170

4.1.2	Creating a New Document Type	170
4.1.3	Modifying a Document Type's Description	173
4.1.4	Modifying a Process Type	173
4.1.4.1	Defining a Process Type's Primary Information	174
4.1.4.2	Defining a Process Type's Templates (Fulfillment Process Types Only) 180	
4.2	Defining Process Type Pipelines	182
4.2.1	Defining Pipeline Determination.....	184
4.2.1.1	Condition Variables for Pipeline Determination.....	184
4.2.2	Creating a Pipeline	185
4.2.2.1	Condition Variables for Pipeline Setup	186
4.2.3	Saving a Pipeline as a Draft	186
4.2.4	Modifying a Pipeline	188
4.2.5	Defining a Pipeline's Monitoring Rules.....	189
4.2.6	Defining Transactions.....	195
4.2.6.1	Creating an Extended Transaction.....	196
4.2.6.2	Creating an Extended Transaction that is Derived from an Abstract Transaction	225
4.2.6.3	Modifying a Transaction.....	229
4.2.6.4	Deleting a Transaction	229
4.2.7	Defining Statuses	230
4.2.7.1	Creating an Extended Status.....	230
4.2.7.2	Modifying an Extended Status	231
4.2.7.3	Deleting an Extended Status.....	232
4.2.7.4	Defining Status Monitoring Rule Definitions	232
4.2.8	Defining Conditions	234
4.2.8.1	Creating a Condition	236
4.2.8.2	Modifying a Condition.....	239
4.2.8.3	Deleting a Condition	241
4.2.8.4	Viewing All Entities Affected by a Condition.....	241
4.2.9	Defining Actions	243
4.2.9.1	Creating an Action.....	244
4.2.9.2	Modifying an Action	249
4.2.9.3	Deleting an Action	249
4.2.10	Defining Service Definitions	250

4.2.10.1	Creating a Service	254
4.2.10.2	Saving a Service as a Draft	257
4.2.10.3	Saving a Service as Another Service	258

5 Configuring User Security

5.1	Defining Users	261
5.1.1	Creating and Modifying a User	262
5.1.2	Setting Up Printer Preferences for a User	265
5.1.3	Deleting a User	268
5.2	Defining User Groups	268
5.2.1	Creating and Modifying a User Group	269
5.2.1.1	Administering User Group Permissions	270
5.2.1.2	Setting Inner Panels as Read-Only	274
5.2.1.3	Setting Up Permissions for Interoperability Servlet	275
5.2.1.4	Setting Up Permissions for APIs	277
5.2.1.5	Viewing the Users Subscribed to a User Group	279
5.2.1.6	Setting Up Printer Preferences for a User Group	279
5.2.2	Deleting a User Group	282
5.3	Defining Teams	282
5.3.1	Creating a Team	282
5.3.2	Modifying a Team	286
5.3.3	Deleting a Team	287
5.4	Defining Data Access Policies	287
5.4.1	Enterprise User Access	288
5.4.2	Buyer User Access	290
5.4.3	Seller User Access	293
5.4.4	Node User Access	293
5.5	Configuring API Security	294
5.5.1	Viewing API Security Resources	295
5.5.2	Creating an API Security Resource	295
5.5.3	Modifying an API Security Resource:	296

6 Configuring System Administration Components

6.1	Defining Purge Criteria	301
6.1.1	Modifying a System Purge Criteria Rule	302

6.2	Defining User Exit Implementations	304
6.2.1	Defining a User Exit	306
6.3	Defining Installation Rules	310
6.4	Defining Agent Criteria Groups	320
6.4.1	Creating an Agent Criteria Group	320
6.4.2	Modifying an Agent Criteria Group	321
6.4.3	Deleting an Agent Criteria Group	322
6.5	Defining Initial Context Factory Codes	322
6.5.1	Creating an Initial Context Factory Code	322
6.5.2	Modifying an Initial Context Factory Code	323
6.5.3	Deleting an Initial Context Factory Code	324
6.6	Defining Health Monitor Rules	324
6.7	Viewing the List of Configured Servers	325
6.7.1	List of Sub Flows or Criteria ID Configured for Server	327
6.8	Defining Application Version.....	327
6.8.1	Creating an Application Version	327
6.8.2	Modifying an Application Version	328
6.8.3	Deleting an Application Version	328

7 Configuring Units of Measure

7.1	Defining Quantity Units of Measure	331
7.1.1	Creating a Unit of Measure for Quantity.....	332
7.1.2	Modifying a Unit of Measure for Quantity	332
7.1.3	Creating a Unit of Measure Conversion Rate for Quantity	332
7.1.4	Modifying a Unit of Measure Conversion Rate for Quantity.....	333
7.1.5	Deleting a Unit of Measure Conversion Rate for Quantity	333
7.1.6	Deleting a Unit of Measure for Quantity	333
7.2	Defining Service Quantity Units of Measure.....	334
7.2.1	Creating a Unit of Measure for Service Quantity.....	334
7.2.2	Modifying a Unit of Measure for Service Quantity	334
7.2.3	Creating a Unit of Measure Conversion Rate for Service Quantity	335
7.2.4	Modifying a Unit of Measure Conversion Rate for Service Quantity.....	335
7.2.5	Deleting a Unit of Measure Conversion Rate for Service Quantity	336
7.2.6	Deleting a Unit of Measure for Service Quantity	336
7.3	Defining Dimension Units of Measure	336

7.3.1	Creating a Unit of Measure for Dimension	336
7.3.2	Modifying a Unit of Measure for Dimension	337
7.3.3	Creating a Unit of Measure Conversion Rate for Dimension	337
7.3.4	Modifying a Unit of Measure Conversion Rate for Dimension	338
7.3.5	Deleting a Unit of Measure Conversion Rate for Dimension	338
7.3.6	Deleting a Unit of Measure for Dimension	338
7.4	Defining Volume Units of Measure	339
7.4.1	Creating a Unit of Measure for Volume	339
7.4.2	Modifying a Unit of Measure for Volume	339
7.4.3	Creating a Unit of Measure Conversion Rate for Volume	340
7.4.4	Modifying a Unit of Measure Conversion Rate for Volume	340
7.4.5	Deleting a Unit of Measure Conversion Rate for Volume	341
7.4.6	Deleting a Unit of Measure for Volume	341
7.5	Defining Weight Units of Measure	341
7.5.1	Creating a Unit of Measure for Weight	341
7.5.2	Modifying a Unit of Measure for Weight	342
7.5.3	Creating a Unit of Measure Conversion Rate for Weight	342
7.5.4	Modifying a Unit of Measure Conversion Rate for Weight	343
7.5.5	Deleting a Unit of Measure Conversion Rate for Weight	343
7.5.6	Deleting a Unit of Measure for Weight	343
7.6	Defining Time Units of Measure	344
7.6.1	Creating a Unit of Measure for Time	344
7.6.2	Modifying a Unit of Measure for Time	344
7.6.3	Creating a Unit of Measure Conversion Rate for Time	345
7.6.4	Modifying a Unit of Measure Conversion Rate for Time	345
7.6.5	Deleting a Unit of Measure Conversion Rate for Time	345
7.6.6	Deleting a Unit of Measure for Time	346

8 Configuring Internationalization Rules

8.1	Defining Country Codes	347
8.1.1	Creating a Country Code Definition	348
8.1.2	Modifying a Country Code Definition	349
8.1.3	Deleting a Country Code Definition	350
8.2	Defining Language Codes	350
8.2.1	Creating a Language Definition	351

8.2.2	Modifying a Language Definition	352
8.2.3	Deleting a Language Definition	352
8.3	Defining Date Formats	352
8.3.1	Creating a Date Format	353
8.3.2	Modifying a Date Format	353
8.3.3	Deleting a Date Format	354
8.4	Defining Time Formats	354
8.4.1	Creating a Time Format	354
8.4.2	Modifying a Time Format	355
8.4.3	Deleting a Time Format	356
8.5	Defining Date/Time Formats	356
8.5.1	Creating a Date/Time Format	356
8.5.2	Modifying a Date/Time Format	357
8.5.3	Deleting a Date/Time Format	358
8.6	Defining Currency Definitions	358
8.6.1	Creating a Currency Definition	359
8.6.2	Modifying a Currency Definition	360
8.6.3	Deleting a Currency Definition	360
8.6.3.1	Defining the Default Reporting Conversion Rate and Euro Currency Code	360
8.7	Defining Currency Conversions	360
8.7.1	Creating a Currency Conversion	361
8.7.2	Modifying a Currency Conversion	363
8.7.3	Deleting a Currency Conversion	363

9 Configuring Presentation Components

9.1	Defining Locales	365
9.1.1	Creating a Locale	366
9.1.2	Modifying a Locale	368
9.1.3	Deleting a Locale	369
9.2	Defining Menus	369
9.2.1	Saving a Menu Group as a New Menu Grouping	369
9.2.2	Modifying Application Menu Details	370
9.2.3	Defining Menu Items	371
9.2.3.1	Creating a Menu Item	371

9.2.4	Modifying a Menu Item	373
9.2.5	Deleting a Menu Item	374
9.3	Defining Resources.....	374
9.3.1	Selling and Fulfillment Foundation Resources	374
9.3.2	Application Consoles Entity	378
9.3.3	Application Consoles Related Entity.....	380
9.3.4	Application Consoles Search View	381
9.3.5	Application Consoles Detail View	382
9.3.6	Application Consoles List View.....	384
9.3.7	Application Consoles API	385
9.3.8	Application Consoles Inner Panel	388
9.3.9	Application Consoles Action	389
9.3.10	Application Consoles Link	391
9.3.11	Application Consoles Icon.....	392
9.3.12	Rich Client Platform Action	393
9.3.13	Rich Client Platform Task.....	393
9.4	Defining Themes	394
9.4.1	Creating a Theme.....	394
9.4.2	Modifying a Theme	395
9.4.3	Deleting a Theme	396
9.5	Defining Custom Common Code Types	396
9.5.1	Creating a Custom Common Code Type.....	396
9.5.2	Modifying a Custom Common Code Type	398
9.5.3	Deleting a Custom Common Code Type.....	398
9.6	Defining Custom Common Code Values	398
9.6.1	Adding Values to a Custom Common Code.....	398
9.6.2	Modifying a Custom Common Code's Values	399
9.6.3	Deleting a Custom Common Code's Values	400
9.7	Defining Custom Error Codes	400
9.7.1	Adding a Custom Error Code.....	401
9.7.2	Modifying a Custom Error Code	401
9.7.3	Deleting a Custom Error Code.....	402

10 Configuring Business Communication Components

10.1	Defining Protocol Codes	403
------	-------------------------------	-----

10.1.1	Creating a Protocol Code	404
10.1.2	Modifying a Protocol Code.....	404
10.1.3	Deleting a Protocol Code	405
10.2	Defining Document Format Codes.....	405
10.2.1	Creating a Document Format Code	405
10.2.2	Modifying a Document Format Code.....	406
10.2.3	Deleting a Document Format Code.....	406
10.3	Defining Business Document Codes	407
10.3.1	Creating a Business Document Code	407
10.3.2	Modifying a Business Document Code.....	408
10.3.3	Deleting a Business Document Code	408
10.3.4	Classifying an Existing Business Document	408

11 Configuring Nomenclature Components

11.1	Defining Nomenclature Codes	412
11.2	Defining Nomenclature Definitions	412
11.2.1	Creating Mappings between Nomenclature Definitions	414
11.3	Defining Nomenclature Configuration	416

12 Configuring Alert Queues

12.1	Creating an Alert Queue	421
12.1.1	Defining How Unresolved Alerts are Handled Based on Size.....	423
12.1.2	Defining How Unassigned Alerts are Handled Based on Time	424
12.1.3	Defining How Unresolved Alerts are Handled Based on Time.....	425
12.1.4	Viewing an Alert Queue's User List.....	426
12.2	Modifying an Alert Queue	426
12.3	Deleting an Alert Queue	427

13 Configuring Region Definitions

13.1	Defining Region Levels	429
13.1.1	Creating a Region Level.....	430
13.1.2	Modifying a Region Level	431
13.1.3	Deleting a Region Level	432
13.2	Defining Region Match Preferences	432

13.2.1	Setting a Region Match Preference	432
13.2.2	Deleting a Region Match Preference	433
13.3	Defining Region Schemas	433
13.3.1	Creating a Region Schema	434
13.3.1.1	Creating a Region	436
13.3.1.2	Modifying a Region	437
13.3.1.3	Deleting a Region	437
13.3.2	Modifying a Region Schema	438
13.3.3	Deleting a Region Schema	438

14 Configuring Devices

14.1	Defining a Device Type	439
14.1.1	Creating a Device Type	440
14.1.2	Modifying a Device Type	441
14.1.3	Deleting a Device Type	442
14.2	Defining a Device Sub Type	442
14.2.1	Creating a Device Sub Type	442
14.2.2	Modifying a Device Sub Type	445
14.2.3	Deleting a Device Sub Type	445
14.3	Defining a Device	445
14.3.1	Creating a Device	446
14.3.1.1	Setting Up a Device Attribute	450
14.3.2	Creating a New Device from a Device	451
14.3.3	Modifying a Device	454
14.3.4	Deleting a Device	454

15 Configuring Prints

15.1	Defining Print Documents	456
15.1.1	Creating a Print Document	456
15.1.2	Modifying a Print Document	458
15.1.3	Deleting a Print Document	459
15.2	Defining Label Formats	459
15.2.1	Creating a Label Format	459
15.2.2	Modifying a Label Format	461
15.2.3	Deleting a Label Format	462

16 Managing Password Policies

17 Configuring Alerts

17.1	Defining Exception Types	466
17.1.1	Creating an Exception Type	466
17.1.2	Modifying an Exception Type.....	468
17.1.3	Deleting an Exception Type.....	469
17.2	Defining Exception Type Role	469
17.2.1	Creating an Exception Type Role	469
17.2.2	Modifying an Exception Type Role.....	470
17.2.3	Deleting an Exception Type Role.....	470
17.3	Defining Organization Exception Type Configuration.....	471
17.3.1	Creating an Exception Type for an Organization.....	471
17.3.2	Modifying an Exception Type for an Organization	472
17.3.3	Creating an Exception Routing Rule	472
17.3.4	Modifying an Exception Routing Rule.....	475
17.3.5	Deleting an Exception Routing Rule.....	475
17.3.6	Deleting an Exception Type for an Organization	476

18 Configuring Data Version Labels

18.1	Creating Configuration Data Version Labels.....	478
18.1.1	Modifying Configuration Data Version Labels	479
18.1.2	Deleting Configuration Data Version Labels	479

19 Configuring Qualified Tag Information

19.1	Defining Qualified Tag Types.....	482
19.1.1	Creating a Qualified Tag Type	482
19.1.2	Modifying a Qualified Tag Type.....	483
19.1.3	Deleting a Qualified Tag Type.....	484
19.2	Defining Qualified Tags	484
19.2.1	Creating a Qualified Tag	484
19.2.2	Modifying a Qualified Tag	485
19.2.3	Deleting a Qualified Tag	486

19.3	Defining Qualified Tag Version Compatibility	486
19.3.1	Configuring Qualified Tag Version Compatibility	486
19.3.2	Modifying Qualified Tag Version Compatibility.....	487
19.3.3	Deleting Qualified Tag Version Compatibility	488

20 Configuring Attribute Postfix Rules

20.1	Defining Attribute Postfix Codes	489
20.1.1	Creating an Attribute Postfix Definition.....	489
20.1.2	Modifying an Attribute Postfix Definition	490
20.1.3	Deleting an Attribute Postfix Definition	491

21 Configuring Analytics

21.1	Defining Analytics Region Usage.....	493
------	--------------------------------------	-----

A Time-Triggered Transaction Reference

A.1	Running Time-Triggered Transactions	497
A.1.1	Steps to Complete Before Scheduling Time-Triggered Transactions	497
A.2	Configuring Communication Between an Agent and a JMS Server	498
A.2.1	Prerequisites	498
A.2.2	Create an Initial Context Factory Code	499
A.2.3	Define the Transaction Information.....	500
A.3	Business Process Time-Triggered Transactions	501
A.3.1	Asynchronous Request Processor.....	502
A.3.2	Case Insensitive Data Loader	504
A.3.3	Change Load Status	506
A.3.4	Change Shipment Status.....	508
A.3.5	Close Delivery Plan.....	510
A.3.6	Close Load	512
A.3.7	Close Manifest	514
A.3.8	Close Order	516
A.3.9	Close Receipts	519
A.3.10	Close Shipment.....	521
A.3.11	Collect Shipment Statistics	523
A.3.12	Consolidate Additional Inventory	526

A.3.13	Consolidate To Shipment	528
A.3.14	Create Catalog Index	532
A.3.15	Create Chained Order	536
A.3.16	Create Derived Order	538
A.3.17	Create Order Invoice	540
A.3.18	Create Shipment Invoice	542
A.3.19	ESP Evaluator	544
A.3.20	Item Based Allocation	547
A.3.21	Mark Load as Trailer Loaded	553
A.3.22	Match Inventory	555
A.3.23	Payment Collection	557
A.3.24	Payment Execution	560
A.3.25	Post Inventory Match	563
A.3.26	Process Order Hold Type	565
A.3.27	Process Work Order Hold Type	568
A.3.28	Publish Negotiation Results	570
A.3.29	Release	572
A.3.30	Route Shipment	575
A.3.31	Schedule	578
A.3.32	Send Invoice	582
A.3.33	Send Item Changes	585
A.3.34	Send Customer Changes	587
A.3.35	Send Order	589
A.3.36	Send Release	591
A.3.37	Start Order Negotiation	593
A.3.38	Synchronize Colony Map	594
A.3.39	Update Best Match Region	596
A.3.40	PopulateOwnershipTransferSummary	599
A.4	Time-Triggered Purge Transactions	600
A.4.1	Purge Strategy	601
A.4.2	Configuring Purge Transaction Log Files	601
A.4.3	Available Purges	602
A.4.3.1	Alert Purge	603
A.4.3.2	Capacity Purge	606
A.4.3.3	Draft Order History Purge	609

A.4.3.4	Draft Order Purge.....	612
A.4.3.5	Delivery Plan Purge	616
A.4.3.6	Export Table Purge	619
A.4.3.7	Import Table Purge.....	621
A.4.3.8	Inventory Audit Purge	624
A.4.3.9	Inventory Purge	627
A.4.3.10	Inventory Supply Temp Purge	630
A.4.3.11	Item Audit Purge.....	632
A.4.3.12	Load History Purge	634
A.4.3.13	Load Purge	637
A.4.3.14	Negotiation History Purge	640
A.4.3.15	Negotiation Purge.....	642
A.4.3.16	Order History Purge	645
A.4.3.17	Order Purge.....	648
A.4.3.18	Order Release Status Purge	655
A.4.3.19	Order Status Audit Purge.....	658
A.4.3.20	Organization Audit Purge.....	660
A.4.3.21	Person Info Purge.....	662
A.4.3.22	Person Info History Purge	665
A.4.3.23	Picklist Purge	668
A.4.3.24	Price List Purge	670
A.4.3.25	Purge Catalog Mass Audits.....	672
A.4.3.26	Receipt History Purge	674
A.4.3.27	Receipt Purge	676
A.4.3.28	Reprocess Error Purge	680
A.4.3.29	Reservation Purge	682
A.4.3.30	Shipment History Purge	684
A.4.3.31	Shipment Purge	686
A.4.3.32	Shipment Statistics Purge.....	690
A.4.3.33	Statistics Purge.....	692
A.4.3.34	User Activity Purge	694
A.4.3.35	User Activity Audit Purge.....	697
A.4.3.36	Work Order History Purge.....	699
A.4.3.37	Work Order Purge	702
A.4.3.38	YFS Audit Purge	705

A.4.3.39	YFSInventoryOwnershipAudit Purge	708
A.4.3.40	Password Reset Request Purge	709
A.4.3.41	User Login Failure Purge.....	711
A.5	Task Queue Syncher Time-Triggered Transactions	713
A.5.1	Load Execution Task Queue Syncher	714
A.5.2	Order Delivery Task Queue Syncher.....	716
A.5.3	Order Fulfillment Task Queue Syncher.....	717
A.5.4	Order Negotiation Task Queue Syncher	718
A.6	Monitors	720
A.6.1	Availability Monitor	720
A.6.2	Exception Monitor.....	722
A.6.3	Inventory Monitor.....	725
A.6.4	Negotiation Monitor	727
A.6.5	Enhanced Order Monitor	729
A.6.6	Enhanced Return Monitor	733
A.6.7	Real-time Availability Monitor.....	737
A.6.8	Shipment Monitor	743
A.6.9	Work Order Monitor	746

B Inventory and Capacity Change Transaction Reference

B.1	The Inventory Change Transaction	749
B.2	The Capacity Change Transaction	752

C Service Builder Nodes and Parameters

C.1	Transport Nodes	755
C.1.1	Database	756
C.1.1.1	Database Sender.....	757
C.1.1.2	Database Receiver.....	759
C.1.2	DCS 6.2 Database	762
C.1.2.1	DDCS 6.2 Database Sender	763
C.1.2.2	DCS 6.2 Database Receiver	764
C.1.3	Component Object Model (COM).....	766
C.1.4	Enterprise Java Beans (EJB)	768
C.1.5	File Input/Output (File I/O).....	769

C.1.5.1	File I/O Sender	770
C.1.5.2	File I/O Receiver	772
C.1.5.3	Enabling EOF Messages in the Application Platform Framework	776
C.1.6	File Transfer Protocol (FTP)	780
C.1.6.1	FTP Sender	780
C.1.6.2	FTP Receiver	784
C.1.7	Hypertext Transport Protocol (HTTP)	788
C.1.8	WebService	791
C.1.9	Synchronous WebLogic and MQSeries	794
C.1.10	WebLogic, MQSeries, and TIBCO JMS Queue	800
C.1.10.1	WebLogic, MQSeries, and TIBCO JMS Queue Sender	801
C.1.10.2	WebLogic, MQSeries, and TIBCO JMS Receiver	809
C.1.11	WebSphere Default Messaging JMS Queue	815
C.1.11.1	WebSphere Default Messaging JMS Queue Sender	815
C.1.11.2	WebSphere Default Messaging JMS Queue Receiver	821
C.1.12	JBoss Default Messaging JMS Queue	827
C.1.12.1	JBoss Default Messaging JMS Queue Sender	827
C.1.12.2	JBoss Default Messaging JMS Queue Receiver	833
C.1.13	Synchronous WebSphere Default Messaging	838
C.1.14	Microsoft Message Queue (MSMQ)	843
C.1.14.1	MSMQ Sender	843
C.1.14.2	MSMQ Receiver	845
C.1.15	Receiver Link Exception Handling	847
C.2	Component Nodes	849
C.2.1	Alert	850
C.2.2	Application Programming Interface (API)	855
C.2.3	Composite Service	858
C.2.4	Condition	859
C.2.5	E-Mail	860
C.2.6	Nomenclature Runtime Component	863
C.2.6.1	Nomenclature Runtime	864
C.2.7	Router	865
C.2.8	Text Translator	866
C.2.9	XSL Translator	868
C.2.10	Defaulting Component	870

C.2.11	Data Security.....	873
C.2.12	Jasper Printer Component	875
C.3	Adapter Nodes	877
C.3.1	Sterling GIS	877

D Text Translator Reference

D.1	Understanding File Formats	881
D.2	Text Translator Components	883
D.2.1	XSD Files	883
D.2.2	Schema Files	884
D.2.3	Defining a Schema File.....	885
D.3	Verifying the Text Translator Setup.....	885
D.3.1	Error Messages	886
D.4	Text Translator XSD Files	889
D.4.1	Text Translator Positional Receiver XSD File	889
D.4.2	Text Translator Delimited Receiver XSD File	896
D.4.3	Text Translator Positional Sender XSD File	899
D.4.4	Text Translator Delimited Sender XSD File	902
D.5	Running the Text Translator.....	904
D.5.1	Specifying the Text Translator Startup Script	904

E Document Types

F Condition Builder Attributes

F.1	Sales Order	911
F.1.1	Order Fulfillment	911
F.1.2	Order Negotiation.....	914
F.1.3	Outbound Shipment.....	915
F.1.4	Sales Order Receipt	916
F.2	Planned Order.....	917
F.2.1	Planned Order Execution	917
F.2.2	Planned Order Negotiation	917
F.3	Return Order	917
F.3.1	Reverse Logistics.....	917

F.3.2	Return Shipment.....	919
F.3.3	Return Receipt.....	920
F.4	Template Order.....	921
F.5	Purchase Order.....	921
F.5.1	Purchase Order Execution.....	921
F.5.2	Purchase Order Negotiation.....	924
F.5.3	Inbound Shipment.....	925
F.5.4	Purchase Order Receipt.....	925
F.6	Transfer Order.....	925
F.6.1	Transfer Order Execution.....	925
F.6.2	Transfer Order Delivery.....	925
F.6.3	Transfer Order Receipt.....	925
F.7	Master Order.....	926
F.7.1	Master Order Fulfillment.....	926
F.8	Load Execution.....	928
F.9	General.....	929
F.10	WMS Putaway.....	931
F.11	WMS Layout Definition.....	931
F.12	WMS Inventory.....	931
F.13	Trailer Loading.....	931
F.14	Task Execution.....	931
F.15	Move Request Execution.....	932
F.16	Manifesting.....	932
F.17	Over Pack Build.....	932
F.18	Count Execution.....	932
F.19	Pack Process.....	934
F.20	Outbound Picking.....	937
F.21	VAS Process.....	937
F.22	Item-Based Allocation (IBA) Order.....	939

Index

Preface

This manual describes how to use the Applications Manager.

Intended Audience

This manual is intended for use by system administrators and managers who need to configure the Selling and Fulfillment Foundation 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 the Applications Manager"

This chapter explains the layout of the Applications Manager, 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 Selling and Fulfillment Foundation. 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 Selling and Fulfillment Foundation, Process Modeling is setting up the Selling and Fulfillment Foundation 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 the Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation 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 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.

Chapter 13, "Configuring Region Definitions"

This chapter explains how you can configure the components that are used by the Selling and Fulfillment Foundation geography engines. This individual components consisting of regions and region levels can be used to create region schemas that can then be used throughout the Selling and Fulfillment Foundation 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.

Chapter 16, "Managing Password Policies"

This chapter provides an overview of password policies and a pointer to *Password Policy Management*. *Password Policy Management* provides more detailed information about managing password policies.

Chapter 17, "Configuring Alerts"

This chapter explains how you can configure alerts to define exception types, exception type roles, and organization exception type configuration.

Chapter 18, "Configuring Data Version Labels"

This chapter explains how you can make it easier to track versions of configuration data, or sets of changes to configuration data using the Configuration Deployment Versioning Tool. It captures changes from a source database, compares and deploys them onto a target database.

Chapter 19, "Configuring Qualified Tag Information"

This chapter explains how you can configure qualified tag and qualified tag types pertaining to rules and common codes.

Chapter 20, "Configuring Attribute Postfix Rules"

This chapter explains how you can define common codes for attribute postfixes.

Chapter 21, "Configuring Analytics"

This chapter explains how you can associate pre-existing region schemas with Data Warehouse Analytics. This association enables reporting on the best region matching mechanism for features such as sourcing and resource pools.

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 Selling and Fulfillment Foundation.

Appendix F, "Condition Builder Attributes"

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

Selling and Fulfillment Foundation Documentation

For more information about the Selling and Fulfillment Foundation components, see the following manuals:

- *Selling and Fulfillment Foundation: Release Notes*
- *Selling and Fulfillment Foundation: Installation Guide*
- *Selling and Fulfillment Foundation: Upgrade Guide*
- *Selling and Fulfillment Foundation: Configuration Deployment Tool Guide*
- *Selling and Fulfillment Foundation: Performance Management Guide*
- *Selling and Fulfillment Foundation: High Availability Guide*
- *Selling and Fulfillment Foundation: System Management Guide*
- *Selling and Fulfillment Foundation: Localization Guide*
- *Selling and Fulfillment Foundation: Customization Basics Guide*
- *Selling and Fulfillment Foundation: Customizing APIs Guide*
- *Selling and Fulfillment Foundation: Customizing Console JSP Interface for End User Guide*
- *Selling and Fulfillment Foundation: Customizing the RCP Interface Guide*
- *Selling and Fulfillment Foundation: Customizing User Interfaces for Mobile Devices Guide*
- *Selling and Fulfillment Foundation: Customizing Web UI Framework Guide*
- *Selling and Fulfillment Foundation: Customizing Swing Interface Guide*
- *Selling and Fulfillment Foundation: Extending the Condition Builder Guide*
- *Selling and Fulfillment Foundation: Extending the Database Guide*
- *Selling and Fulfillment Foundation: Extending Transactions Guide*

- *Selling and Fulfillment Foundation: Using Sterling RCP Extensibility Tool Guide*
- *Selling and Fulfillment Foundation: Integration Guide*
- *Selling and Fulfillment Foundation: Product Concepts Guide*
- *Sterling Warehouse Management™ System: Concepts Guide*
- *Selling and Fulfillment Foundation: Application Platform Configuration Guide*
- *Sterling Distributed Order Management™: Configuration Guide*
- *Sterling Supply Collaboration: Configuration Guide*
- *Sterling Global Inventory Visibility™: Configuration Guide*
- *Catalog Management™: Configuration Guide*
- *Sterling Logistics Management: Configuration Guide*
- *Sterling Reverse Logistics™: Configuration Guide*
- *Sterling Warehouse Management System: Configuration Guide*
- *Selling and Fulfillment Foundation: Application Platform User Guide*
- *Sterling Distributed Order Management: User Guide*
- *Sterling Supply Collaboration: User Guide*
- *Sterling Global Inventory Visibility: User Guide*
- *Sterling Logistics Management: User Guide*
- *Sterling Reverse Logistics: User Guide*
- *Sterling Warehouse Management System: User Guide*
- *Selling and Fulfillment Foundation: Mobile Application User Guide*
- *Selling and Fulfillment Foundation: Business Intelligence Guide*
- *Selling and Fulfillment Foundation: Javadocs*
- *Sterling Selling and Fulfillment Suite™: Glossary*
- *Parcel Carrier: Adapter Guide*
- *Selling and Fulfillment Foundation: Multitenant Enterprise Guide*
- *Selling and Fulfillment Foundation: Password Policy Management Guide*

- *Selling and Fulfillment Foundation: Properties Guide*
- *Selling and Fulfillment Foundation: Catalog Management Concepts Guide*
- *Selling and Fulfillment Foundation: Pricing Concepts Guide*
- *Business Center: Item Administration Guide*
- *Business Center: Pricing Administration Guide*
- *Business Center: Customization Guide*
- *Business Center: Localization Guide*

Conventions

The following conventions may be used in this manual:

Convention	Meaning
. . .	Ellipsis represents information that has been omitted.
< >	Angle brackets indicate user-supplied input.
mono-spaced text	Mono-spaced text indicates a file name, directory path, attribute name, or an inline code example or command.
/ 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.
<INSTALL_DIR>	User-supplied location of the Selling and Fulfillment Foundation installation directory. This is only applicable for Release 8.0 or later.
<INSTALL_DIR_OLD>	User-supplied location of the Selling and Fulfillment Foundation installation directory (for Release 8.0 or later). Note: This is applicable only for users upgrading from Release 8.0 or later.
<YANTRA_HOME>	User-supplied location of the Sterling Supply Chain Applications installation directory. This is only applicable for Releases 7.7, 7.9, and 7.11.

Convention	Meaning
<YANTRA_HOME_OLD>	User-supplied location of the Sterling Supply Chain Applications installation directory (for Releases 7.7, 7.9, or 7.11). Note: This is applicable only for users upgrading from Releases 7.7, 7.9, or 7.11.
<YFS_HOME>	For Releases 7.3, 7.5, and 7.5 SP1, this is the user-supplied location of the Sterling Supply Chain Applications installation directory. For Releases 7.7, 7.9, and 7.11, this is the user-supplied location of the <YANTRA_HOME>/Runtime directory. For Release 8.0 or above, the <YANTRA_HOME>/Runtime directory is no longer used and this is the same location as <INSTALL_DIR>.
<YFS_HOME_OLD>	This is the <YANTRA_HOME>/Runtime directory for Releases 7.7, 7.9, or 7.11. Note: This is only applicable for users upgrading from Releases 7.7, 7.9, or 7.11.
<ANALYTICS_HOME>	User-supplied location of the Sterling Analytics installation directory. Note: This convention is used only in the <i>Selling and Fulfillment Foundation: Business Intelligence Guide</i> .
<COGNOS_HOME>	User-supplied location of the IBM Cognos 8 Business Intelligence installation directory. Note: This convention is used only in the <i>Selling and Fulfillment Foundation: Business Intelligence Guide</i> .
<MQ_JAVA_INSTALL_PATH>	User-supplied location of the IBM WebSphere® MQ Java components installation directory. Note: This convention is used only in the <i>Selling and Fulfillment Foundation: System Management and Administration Guide</i> .
<DB>	Refers to Oracle®, IBM DB2®, or Microsoft SQL Server® depending on the database server.
<DB_TYPE>	Depending on the database used, considers the value oracle, db2, or sqlserver.

Note: The Selling and Fulfillment Foundation documentation set uses the following conventions in the context of the product name:

- Yantra is used for Release 7.7 and earlier.
- Sterling Supply Chain Applications is used for Releases 7.9 and 7.11.
- Sterling Multi-Channel Fulfillment Solution is used for Releases 8.0 and 8.2.
- Selling and Fulfillment Foundation is used for Release 8.5.

Introduction

This book concentrates on the rules and setup configurations that make up the Applications Manager. This book is intended for both Hub and Enterprise administrators using the Applications Manager to set up the Selling and Fulfillment Foundation environment. Business analysts should also use this book to plan appropriate business practices as they pertain to Selling and Fulfillment Foundation. Programmers should refer to the *Selling and Fulfillment Foundation: Customization Basics Guide* for information about extending Selling and Fulfillment Foundation. System Integrators should refer to the *Selling and Fulfillment Foundation: Integration Guide* for information about integrating external applications with Selling and Fulfillment Foundation.

Important: This book assumes that you have read and are familiar with the concepts and business functionality detailed in the *Selling and Fulfillment Foundation: Product Concepts Guide*.

The Applications Manager is a collection of all the rules and setup configurations necessary to implement Selling and Fulfillment Foundation, organized in such a way that configuration can be done for each business application separately. The following business applications can be configured within the Applications Manager:

- Distributed Order Management
- Global Inventory Visibility
- Catalog Management
- Logistics Management

- Supply Collaboration
- Reverse Logistics
- Warehouse Management
- Application Platform

1.1 Business Models

There is no single business model that encompasses the environment in which all the Selling and Fulfillment Foundation applications can be used. Therefore, there is no single way to configure your Selling and Fulfillment Foundation 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 the Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation. 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.

Selling and Fulfillment Foundation 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 Application Platform Configuration

Selling and Fulfillment Foundation is a collection of common components used across the application. These components are the infrastructure upon which all other business application modules in Selling and Fulfillment Foundation are built.

In the Applications Manager you can use the Application Platform configuration grouping to establish the following aspects of Selling and Fulfillment Foundation for your business applications:

- [Participant Modeling](#)
- [Process Modeling](#)
- [Security](#)
- [System Administration](#)
- [Unit of Measure](#)
- [Internationalization](#)
- [Presentation](#)
- [Nomenclature](#)
- [Queue Management](#)
- [Region Definition](#)
- [Alert Management](#)

1.2.1 Participant Modeling

The Applications Manager'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 Process Modeling

The 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 life cycle of that process type. For more information about the 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 Application Console and the

Applications Manager. 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 teams. 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 Selling and Fulfillment Foundation 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. Selling and Fulfillment Foundation provides unit of measure classifications for dimension, volume, weight, and time. In addition to defining new units of measure, Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation 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 Queue Management

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. 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 Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation business application models whenever geography is considered. For more information about Region Definition, see [Chapter 13, "Configuring Region Definitions"](#).

1.2.12 Alert Management

Alert management allows a user to create new exception types, organization exception types, queues, and exception routing rules. Exception types are maintained at the Hub level and contain information such as which queue the exception is assigned to, the default priority of the exception, and the high priority threshold. The lower the priority number, the higher the alert's priority is. The high priority threshold

value is used to determine what alerts are of a high priority. Any alert with a priority number lower than that of the high priority threshold is considered high priority. There are also several PCA specific fields that can be configured here, such as Consolidation Hook, Resolution Hook, and Resolution Wizard.

When overriding an exception type for a particular organization in order to enable it to route to a different queue or be raised with a different priority, you can create a new organization exception type. On the organization exception type, you can configure complex routing rules to determine what queue the exception is placed in upon creation. The organization exception type can also be configured to send an email notification if an exception of its type is raised. These organization exception types are maintained at the organization level. You can only override an exception once for a particular organization.

For more information about alerts, see [Chapter 17, "Configuring Alerts"](#).

Navigating the Applications Manager

This chapter discusses the layout of the Applications Manager, actions you can perform throughout the application, and important concepts you should be aware of before using the application.

2.1 Starting the Applications Manager

To access the Applications Manager:

1. Point your browser to

`http://<hostname>:<portname>/smcfs/console/start.jsp`

where,

- `hostname` is the computer name or IP address of the computer where Selling and Fulfillment Foundation is installed.
- `portnumber` is the listening port of the computer where Selling and Fulfillment Foundation is installed.

The browser displays the Sign In window.

2. Enter your login ID and password and choose the Sign In button. The Console Home Page is displayed.
3. From the menu bar, choose Configuration > Launch Applications Manager. The Applications Manager opens in a new window.

Note: Additionally, enterprise users who maintain an enterprise can access the Applications Manager by means of `http://<Selling and Fulfillment Foundation installation server>/smcfs/console/login.jsp`.

Note: If both the Applications Manager and the monitor in the System Management Console are opened at the same time, and if a dialogue window is opened in either application, the other stops responding to user input until that dialogue window is closed. This is due to a bug in the Java platform.

2.2 The Applications Manager Layout

The Applications Manager is a graphical user interface that can be used to configure different aspects of Selling and Fulfillment Foundation. The different configurations are defined by logical groupings called applications that can be accessed from the Applications Manager's menu bar.

Figure 2–1 Applications Menu



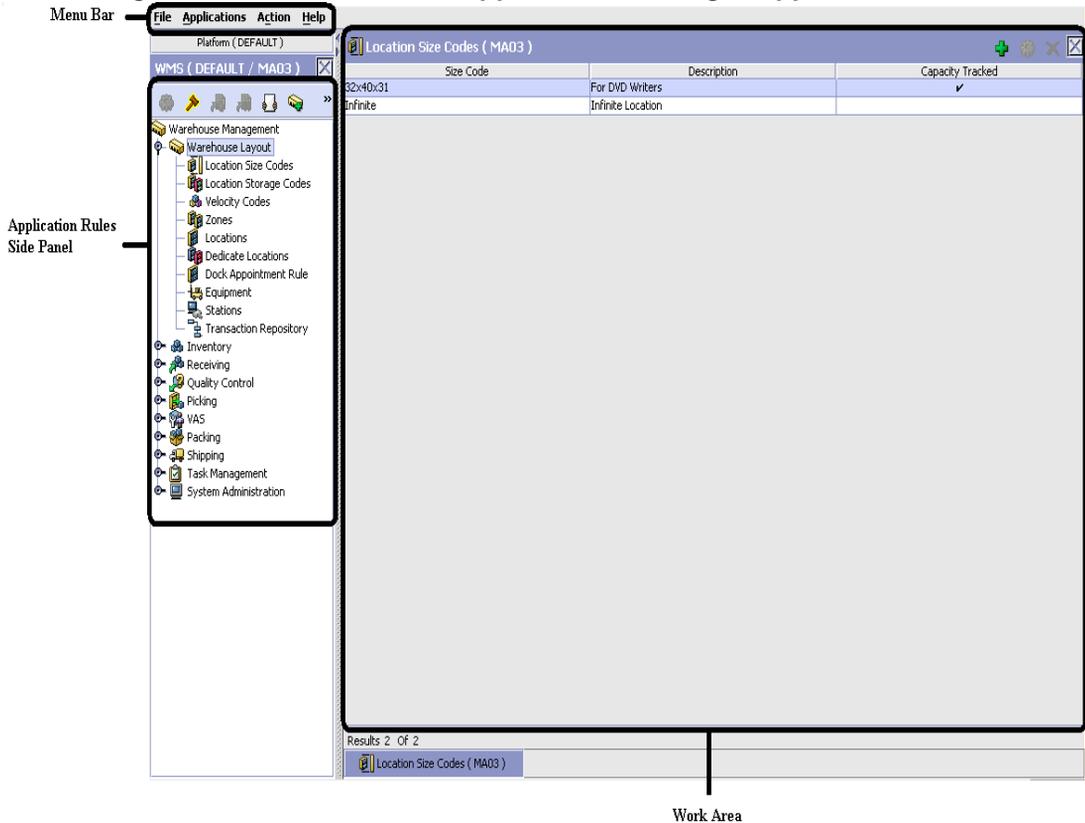
Each application focuses on a particular aspect of Selling and Fulfillment Foundation and contains all of the rules, common codes, and settings necessary for Selling and Fulfillment Foundation to work in a real-world business setting.

The following applications can be configured in this version of Selling and Fulfillment Foundation:

- Distributed Order Management
- Global Inventory Visibility
- Catalog Management
- Logistics Management
- Supply Collaboration
- Reverse Logistics
- Warehouse Management
- Application Platform

When you select the application that you want to configure, the Applications Manager 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.

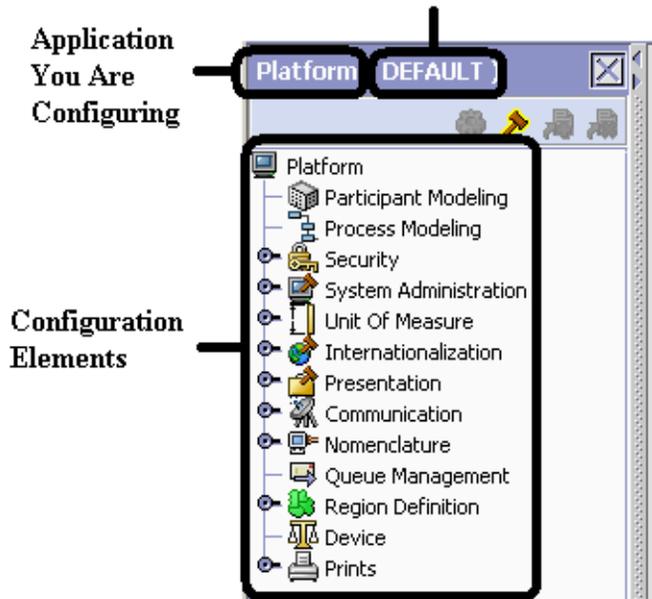
Figure 2–2 The Standard Applications Manager Application Interface



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 Organization You Are Defining Rules For



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 displays in the work area.

2.2.1.2 Determining Inheritance

In Selling and Fulfillment Foundation, 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"](#).

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 Selling and Fulfillment Foundation 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
- Pricing Organization - Configuration groups that are associated with the organization(s) that maintains the pricing within the Hub environment
- Organization - Configuration groups that are associated with any organization within the Hub environment

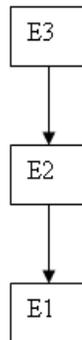
Note: The Applications Manager does not load configuration data and permissions based on Data Access Policies that are described in the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

Enhanced Inheritance for Process Models

An Enterprise can inherit the configurations of the following entities from other Enterprises:

- Pipelines
- User Exits
- Services
- Actions
- Conditions
- Statuses
- Transactions
- Events

When an Enterprise inherits these entities from some other Enterprise, the current Enterprise can view the configurations that are inherited from all other Enterprises (including the Hub) in the inheritance hierarchy. In addition, the current Enterprise can view the configurations that are defined for the Hub. For example, consider the following inheritance hierarchy:



In this hierarchy, Enterprise E1 is inheriting from Enterprise E2, which in turn is inheriting from Enterprise E3. Enterprise E1 can view the configurations that are defined for Enterprise E2 and Enterprise E3. In addition, Enterprise E1 can view the configurations that are defined for the Hub.

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.

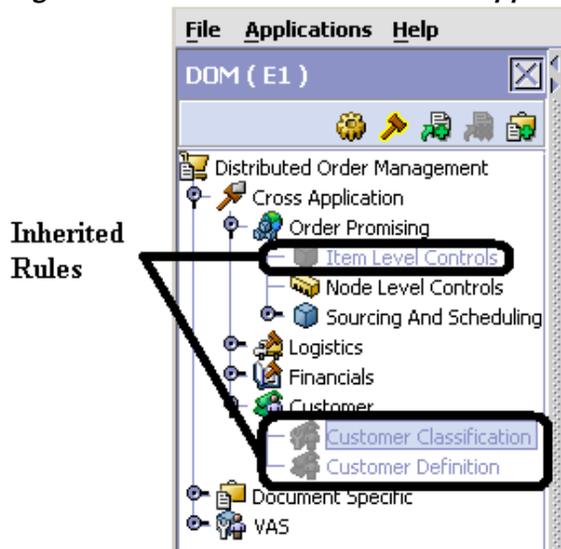
Table 2–1 Organization Level Rules

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.
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.
Pricing Organizations	Organizations that are designated as pricing organizations can modify configuration groups at the pricing 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.

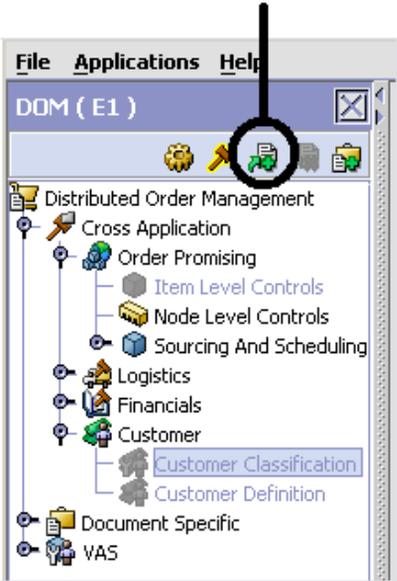
The application rules side panel displays rules that have been inherited as grayed out.

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



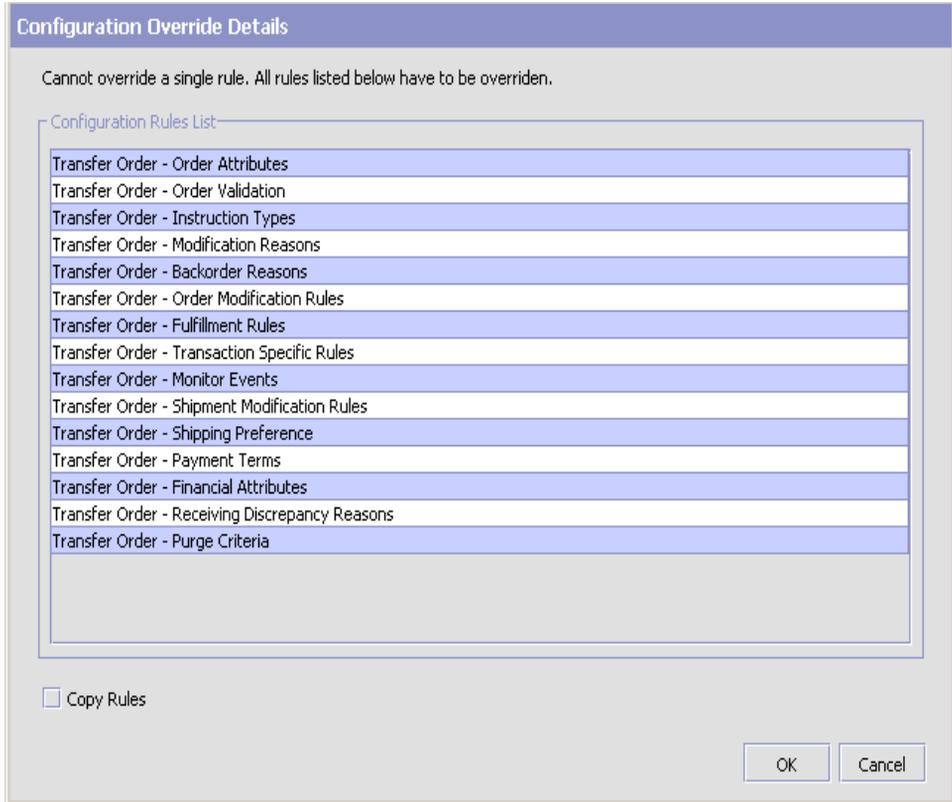
As stated in the table above, depending on the organization you are logged in as, you may be able to override some inherited rules. If a rule can be overridden, the Override Configuration icon becomes available in the application rule side panel when you highlight the rule.

Figure 2–5 Override Configuration Icon
Override Configuration Icon is Available



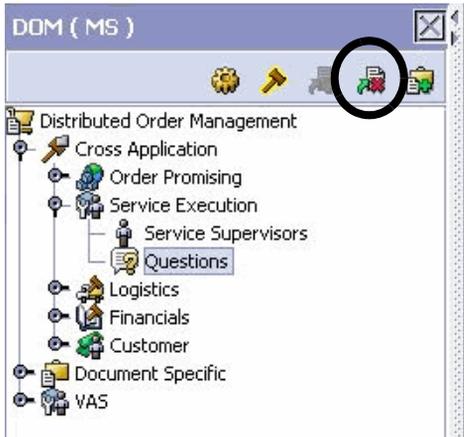
When you choose to override a rule you also override any other rules in the configuration group the rule you are overriding is associated with. When you choose the Override Configuration icon the Configuration Override Details pop-up window displays. This window provides the list of rules that are overridden.

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



When you select the Give Back Configuration Ownership Icon, the Configuration Override Details pop-up window displays. This window provides the list of rules that are 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.

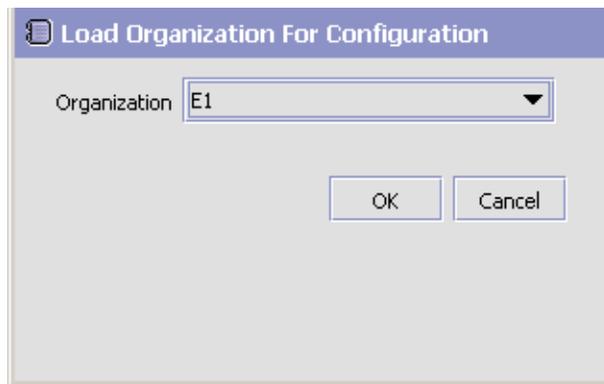
2.2.1.3 Loading Another Organization’s Rules

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 displays.



2. From Organization, select the organization that you want to work with.
3. Choose OK. The organization's rules display 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

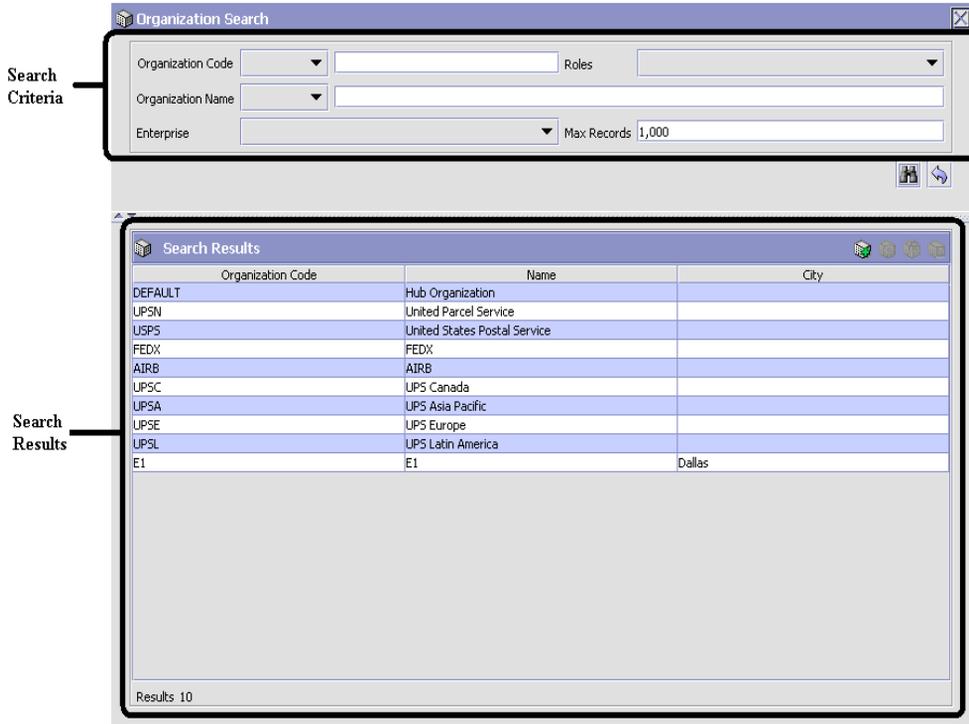
The work area is the main area in which different configuration screens appear. The following are the main types of screens that you can be seen in the work area:

- [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 Applications Manager may display a basic list window of the rules and codes that have previously been configured.

Figure 2–9 List Window Example

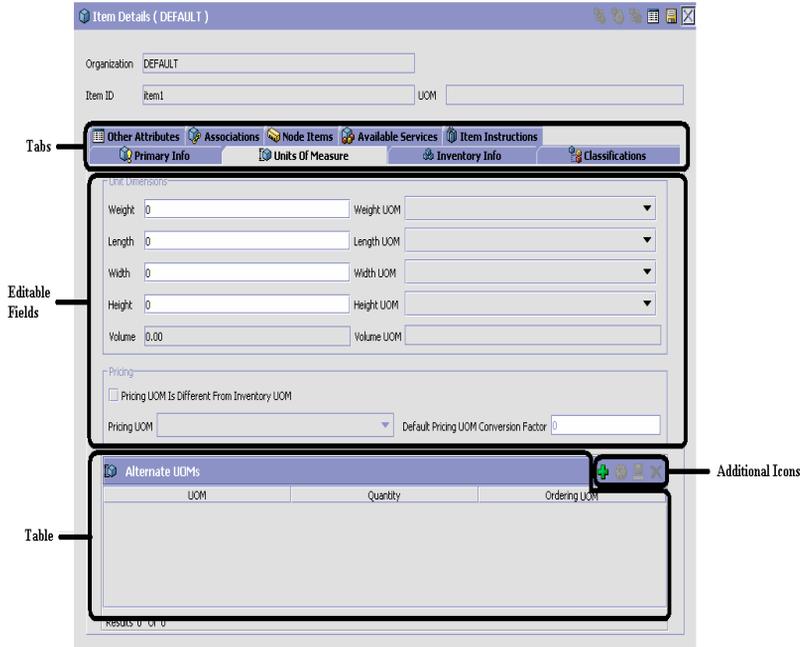
UOM Code	UOM Description
CM	Centimeter
FEET	Feet
IN	Inch
KM	Kilometer
METER	Meter
MILE	Mile

Results 6 Of 6

2.2.2.3 Details Window

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

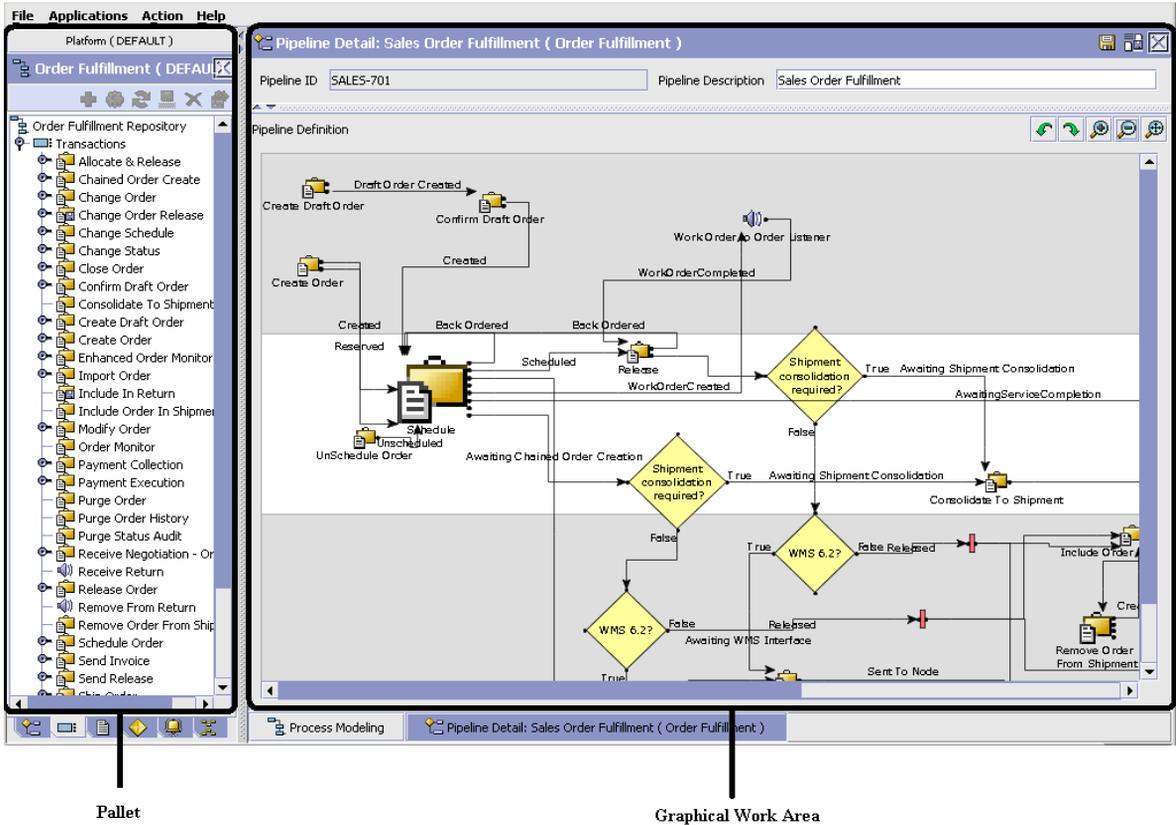
Figure 2–10 Details Window Example



2.2.2.4 Drag and Drop Window

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

Figure 2–11 Drag and Drop Window Example



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

To connect one component to another, you must drag the mouse from the outgoing port of a component until it forms a connecting line with the incoming port of another component. The links between components can be set up either horizontally or vertically.

To delete components or links, right-click on 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 press and hold the CTRL key while dragging a component, the component is copied within the graphical work area.

2.3 Actions Available in the Applications Manager

The following actions can be performed throughout the Applications Manager:

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

2.3.1 Using the Applications Manager's Lookup Functionality

Throughout the Applications Manager 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 displays in a pop-up screen for you to modify.

Figure 2–12 Lookup Icon Example



The information that displays 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 displays on save if the user has entered an invalid value.

When you use a lookup for a particular field in the Applications Manager, 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, Master 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 displays displaying a list of all of the document types associated with the application you are working in.

Figure 2–13 Associated Document Types Window

The screenshot shows a window titled "Associated Document Types" with a table containing the following data:

Document Type	Description
0004	Template Order
0006	Transfer Order
0001	Sales Order
0007	Master Order

Results 4 of 4

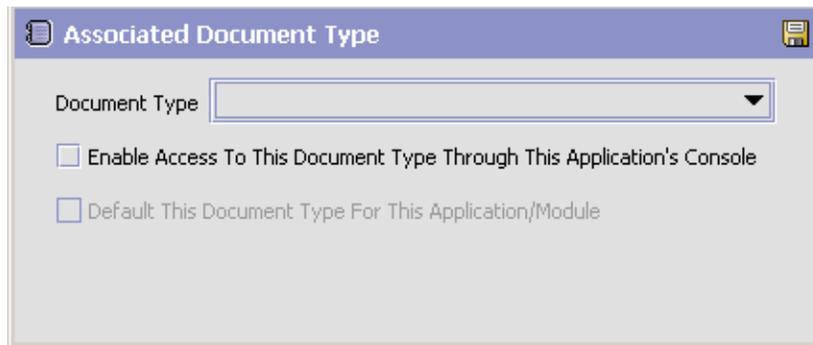
2.3.2.1 Adding a Document Type to an Application

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 displays.



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 Application's Console.
4. Choose .

2.3.3 Viewing the User Logged into the Applications Manager

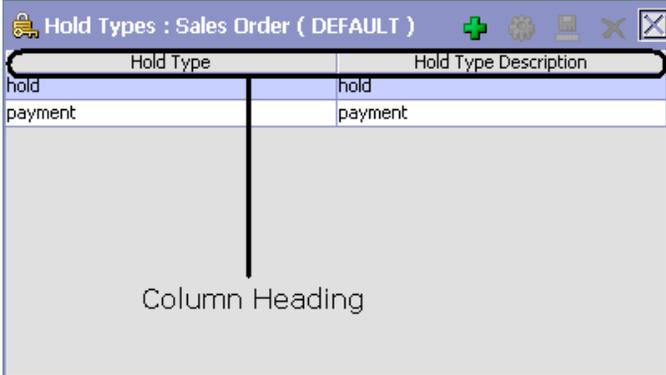
You can view the user logged into the Applications Manager and their locale at any time. To view this information, move your mouse over the User icon and Locale icons in the bottom right-hand corner of the application to display the tool tips.

2.3.4 Using Lists and List Filtering

When viewing any list in the Applications Manager, it is possible to filter the contents of the list based in criteria that you define. Filtering is

accomplished by right-clicking anywhere on the list's column headings and using the Table Filter Editor associated with the list.

Figure 2–14 Column Headings in a List



Hold Type	Hold Type Description
hold	hold
payment	payment

Column Heading

Figure 2–15 Table Filter Editor Window Example

Table 2–2 Table Filter Editor Window

Field	Description
Apply To Existing Records	Checking this box applies a new filter set of results that have been previously filtered instead of the whole set.
Max Records	Specify the maximum number of records that are to be returned from a filter. The default number is 100
Dynamic Fields	Fields such as "Hold Type" and "Hold Type Description" in Figure 2–15 are dynamically populated based on the list you are currently viewing. These fields can be searched using text strings combined with criteria such as Is , Starts With , or Contains .

Important: Search strings are case sensitive. For example, "Item" does not return the same values as "item".

2.3.5 Date and Time Entry

Date fields through the Applications Manager 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 2–16 Calendar Icon example



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

Figure 2–17 Time Field example



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

2.3.6 Using On-Line Help

You can access the Selling and Fulfillment Foundation Online Help through Help > Online Help.

2.3.7 Troubleshooting Errors

You can view the description and cause of any error raised in Selling and Fulfillment Foundation, as well as the actions to troubleshoot it.

To view the Selling and Fulfillment Foundation system error descriptions:

1. From the menu bar, choose Help > Troubleshooting. The Error Search window displays.

2. Enter the applicable search criteria and choose . A list of error codes and their descriptions display.
3. Choose  to view the cause of the error and action to troubleshoot it.

2.3.8 Using Special Characters

Throughout the Applications Manager there may be instances where you need to use special characters in data entry. For information about the use of special characters in Selling and Fulfillment Foundation, see the *Selling and Fulfillment Foundation: Customization Basics Guide*.

Configuring Participants

Trading partners using Selling and Fulfillment Foundation 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 displays 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
DUNS Number	Enter a unique nine-digit identification sequence, which provides unique identifiers of single business entities. Selling and Fulfillment Foundation 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	Choose the organization's geographic location from the drop-down list.
Parent Organization	Choose the parent organization from the drop-down list.
Organization Is An Enterprise	<p>Check this box to configure the organization as an Enterprise.</p> <p>Note: The Primary Enterprise drop-down list is disabled when you select this checkbox.</p>
Assigned Colony Id	<p>From the drop-down list, select the Colony ID or DEFAULT Colony ID. The DEFAULT Colony ID is installed automatically with Selling and Fulfillment Foundation. If you do not make a selection, DEFAULT is assumed.</p> <p>Note: This selection is available only if the Organization Is An Enterprise check box is enabled and if this organization is part of a multi-schema deployment.</p>

Table 3–1 Create Organization Popup

Field	Description
Primary Enterprise	<p>Select the appropriate primary enterprise from the drop-down list when the 'Organization Is An Enterprise' checkbox is not selected.</p> <p>This Primary Enterprise is the default enterprise displayed on the entry point for 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 for a specific organization, the rules are always retrieved for the Primary Enterprise of that organization.</p> <p>Note: The 'Primary Enterprise must be specified.' error is thrown while saving, if neither 'Organization Is An Enterprise' checkbox nor an appropriate primary enterprise is selected from the drop-down list.</p>
Organization is a Fulfillment Node	<p>Check this box to indicate that the organization is a fulfillment node.</p> <p>In Node, select the node type from the drop-down list. Choose  to create a node type. For more information about creating node types, see Section 3.3, "Creating Node Types".</p>
Organization is a Yard	<p>Check this box to indicate that the organization is a yard.</p>
Address	<p>Enter the address of the organization's corporate headquarters. This information is mandatory.</p> <p>Choose  to enter the 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> <p>Latitude and Longitude need to be entered using decimal format with a range of -90 to +90 for latitude and -180 to +180 for longitude.</p>

Note: A user cannot create a node unless the node that is being created will be automatically accessible to the user after creation.

The newly created node will be automatically accessible to the user in the following scenarios:

- The user who creates the node is an Enterprise user, and
 - no team is associated to the user
 - the user's team's ShipNode access is set to "All Nodes"
 - the user's team's ShipNode access is set to "User's Node", and the user's enterprise is the parent of the node that is being created
 - the user's team's ShipNode access is set to "Nodes Accessible to Team Creator", and the user who has created the team has automatic access to the newly created node
- The user who creates the node is a Node user, the user's team's ShipNode access is set to "Nodes Accessible to Team Creator", and the user who has created the team has automatic access to the newly created node.

Any user who has access to a restricted set of nodes in the user's team configuration cannot create a ship node. Also, if the user is a Node user, and no team is associated to that user, the user cannot create a ship node.

You can use the Organization Details window for:

- [Defining an Organization's Primary Information](#)
- [Assigning the Organization's Roles and Participant Associations](#)
- [Assigning a Yard's Roles and Participant Associations](#)
- [Defining Communication Protocols](#)
- [Defining an Organization's Payment Information](#)

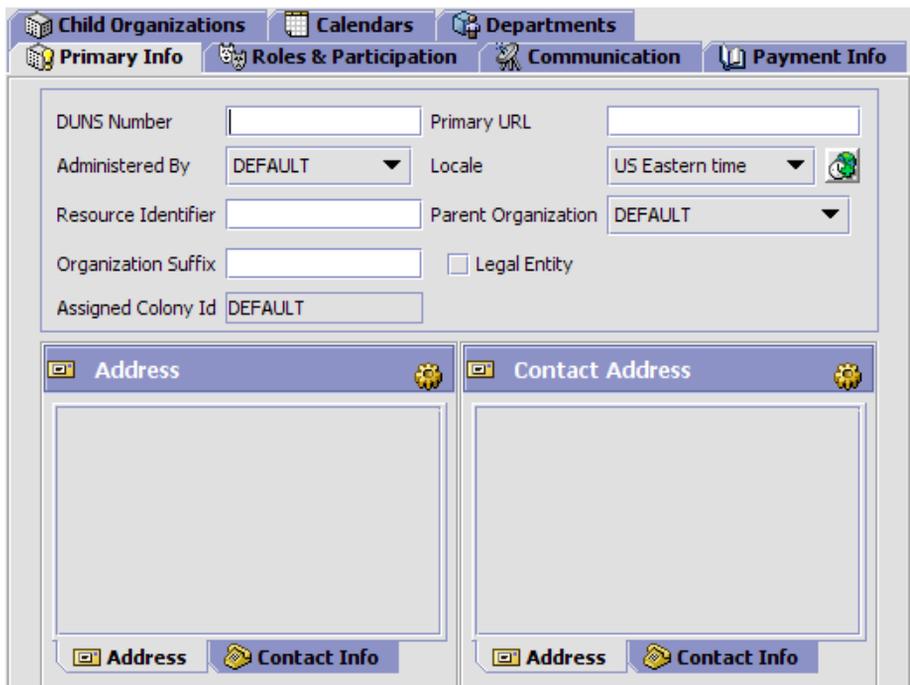
- [Viewing an Organization's Child Organizations](#)
- [Defining an Organization's Calendars](#)

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 .



The screenshot shows the 'Primary Info' tab in the Organization Details window. The interface includes a top navigation bar with tabs for 'Child Organizations', 'Calendars', and 'Departments'. Below this, the 'Primary Info' tab is active, with other tabs for 'Roles & Participation', 'Communication', and 'Payment Info'. The main form area contains several fields: 'DUNS Number' (text input), 'Administered By' (dropdown menu set to 'DEFAULT'), 'Resource Identifier' (text input), 'Organization Suffix' (text input), and 'Assigned Colony Id' (dropdown menu set to 'DEFAULT'). On the right side, there are 'Primary URL' (text input), 'Locale' (dropdown menu set to 'US Eastern time' with a globe icon), and 'Parent Organization' (dropdown menu set to 'DEFAULT'). A 'Legal Entity' checkbox is also present. Below the main form, there are two sections: 'Address' and 'Contact Address', each with a gear icon and a sub-section for 'Address' and 'Contact Info'.

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 Applications Manager.
Locale	Select the organization's geographic location.
Resource Identifier	Enter the Resource Identifier for the organization.
Parent Organization	Choose in the organization hierarchy who is the parent organization.
Organization Suffix	Enter the organization suffix for the organization. The organization suffix is used when creating a buyer organization for a business customer.

Table 3–2 Primary Info Tab

Field	Description
Legal Entity	<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>
Assigned Colony Id	<p>A read-only field that displays the Colony ID that was selected in the Create Organization screen. If no selection was made, DEFAULT is assumed.</p> <p>Note: This field displays only in a multi-schema deployment.</p>

Table 3–2 Primary Info Tab

Field	Description
Address	<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> <p>Latitude and Longitude need to be entered using decimal format with a range of -90 to +90 for latitude and -180 to +180 for longitude.</p>
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 are used to plot the node on the Fulfillment Network Model. For more information about the Fulfillment Network Model, see the <i>Sterling Distributed Order Management: Configuration Guide</i>.</p> <p>Latitude and Longitude need to be entered using decimal format with a range of -90 to +90 for latitude and -180 to +180 for longitude.</p>

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. Selling and Fulfillment Foundation supports the following organization roles:

- Buyer
- Carrier

- Enterprise
- Hub
- Node
- Seller

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.

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.

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 Selling and Fulfillment Foundation controls the flow of order and logistics documents and is considered the owner of the various business documents throughout Selling and Fulfillment Foundation.

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.

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 Selling and Fulfillment Foundation. 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.

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.

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.

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.



The screenshot shows the 'Enterprise Attributes' configuration window. It has two tabs: 'Enterprise Attributes' (selected) and 'Advanced Attributes'. The 'Enterprise Attributes' tab contains the following fields:

- Enterprise Name:** A text input field.
- Inherit Configuration From Enterprise:** A dropdown menu.
- UCC Prefix:** A text input field.
- Password Policy:** A dropdown menu.
- Suppress Chained Order Creation Even If Seller Requires Chained Order:** A checkbox.

2. In Enterprise Name, enter the name of the Enterprise you are configuring.
3. If you 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.

4. In UCC Prefix, enter the UCC code to be used when identifying the Enterprise on a shipment container marking (SCM).
5. In Password Policy, select the password policy that you want to associate with the enterprise.

For additional information about password policies, see the *Selling and Fulfillment Foundation: Password Policy Management Guide*.

- If you want to suppress chained order creation even if the seller requires chained orders, select the Suppress Chained Order Creation Even If Seller Requires Chained Order check box.

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 displays.

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 .

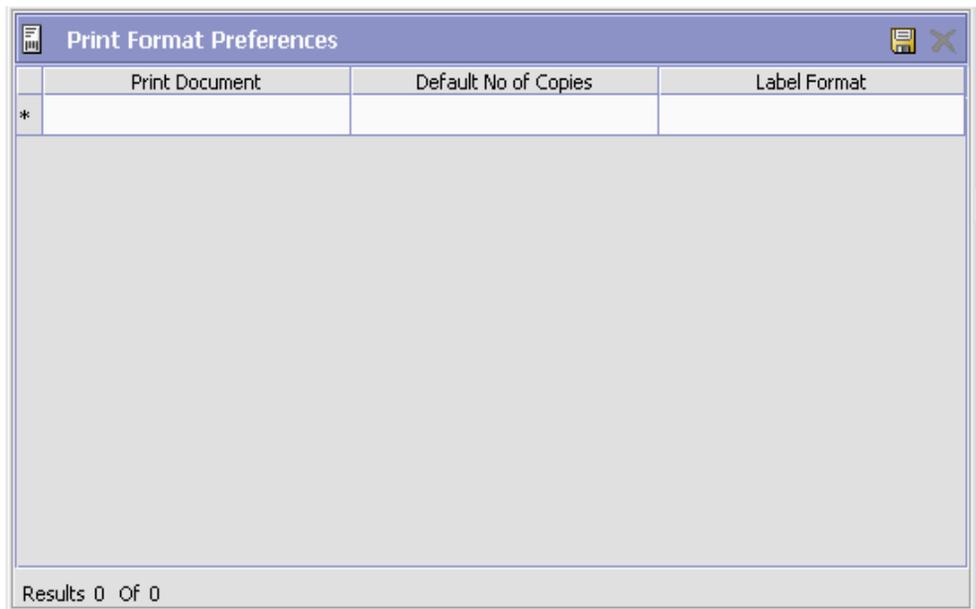


Table 3–3 Enterprise Attributes Print Format Preferences Tab

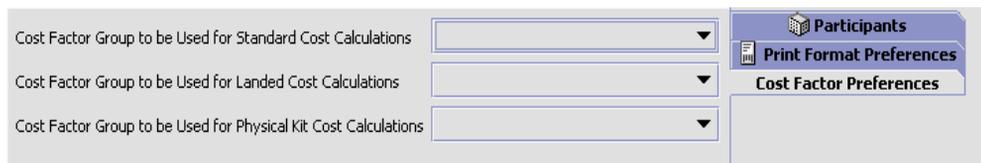
Field	Description
Print Document	From the drop down, select the print document for which you want to configure preferences.
Default No of Copies	Enter the number of copies that should be printed.
Label Format	Select which label format should be used with this document type.

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.

3.1.2.1.7 Defining an Enterprise’s Inventory Monitoring Rules

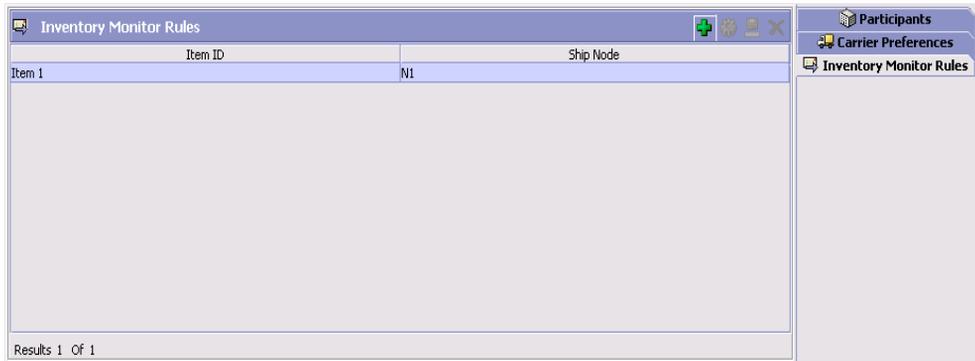
If an item is tracked at an Enterprise’s node, you can set up minimum and maximum inventory levels using inventory monitoring rules for an item. When the item reaches one of these levels, a corresponding event is raised if an action is configured. That event can then be configured to trigger an alert, if applicable.

For example, you can set a minimum quantity of 75 for a particular item. When the inventory starts to drop and reaches the minimum level of 75, you can configure the minimum action to raise the Send E-Mail action. This sends an e-mail message to notify a manager of the decreasing inventory level.)

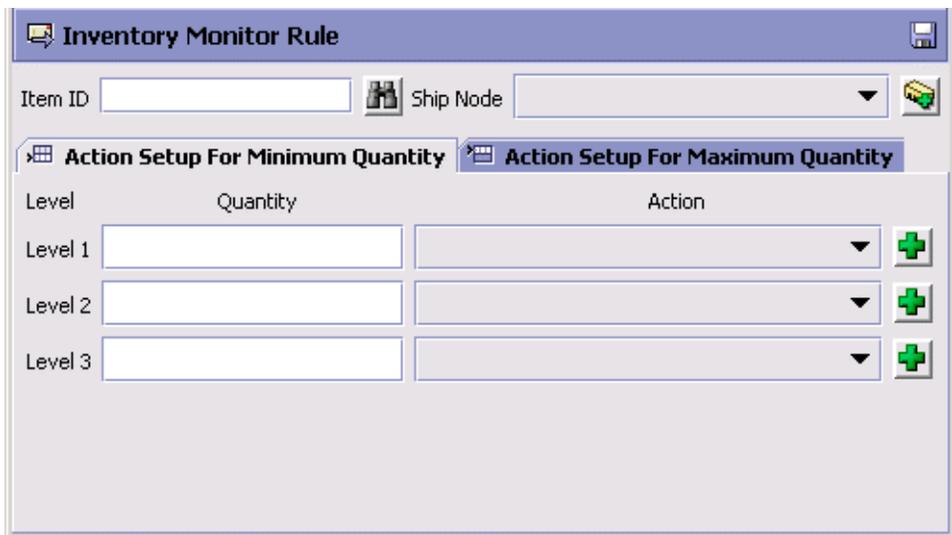
You can add, modify, and delete inventory monitoring rules.

To add an inventory monitoring rule to an Enterprise:

1. From the Roles & Participation tab in the Organization Details window, choose Enterprise Attributes.
2. Choose the Inventory Monitor Rules tab. The Inventory Monitor Rules list displays.



3. Choose . The Inventory Monitor Rule pop-up window displays.
4. Enter information in the applicable fields. Refer to for field value descriptions.
5. Choose .



To modify an Enterprise's inventory monitoring rule:

1. From the Roles & Participation tab in the Organization Details window, choose Enterprise Attributes.
2. Choose the Inventory Monitor Rules tab. The Inventory Monitor Rules list displays.
3. Select the applicable inventory monitor rule and choose . The Inventory Monitor Rule pop-up window displays.
4. Enter information in the applicable fields. Refer to for field value descriptions.
5. Choose .

To delete an Enterprise's inventory monitoring rule:

1. From the Roles & Participation tab in the Organization Details window, choose Enterprise Attributes.
2. Choose the Inventory Monitor Rules tab. The Inventory Monitor Rules list displays.
3. Select the applicable inventory monitor rule and choose .

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.



Seller Attributes	Enterprise Attributes	Advanced Attributes
<input checked="" type="checkbox"/> Payment Processing Required	Default Payment Rule	DEFAULT
<input checked="" type="checkbox"/> Requires Chained Orders	Default Price Program	

Note: The Default Price Program drop-down list is displayed only when you use the legacy pricing service. For more information about enabling the legacy pricing service, refer to [Defining Installation Rules](#).

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 .

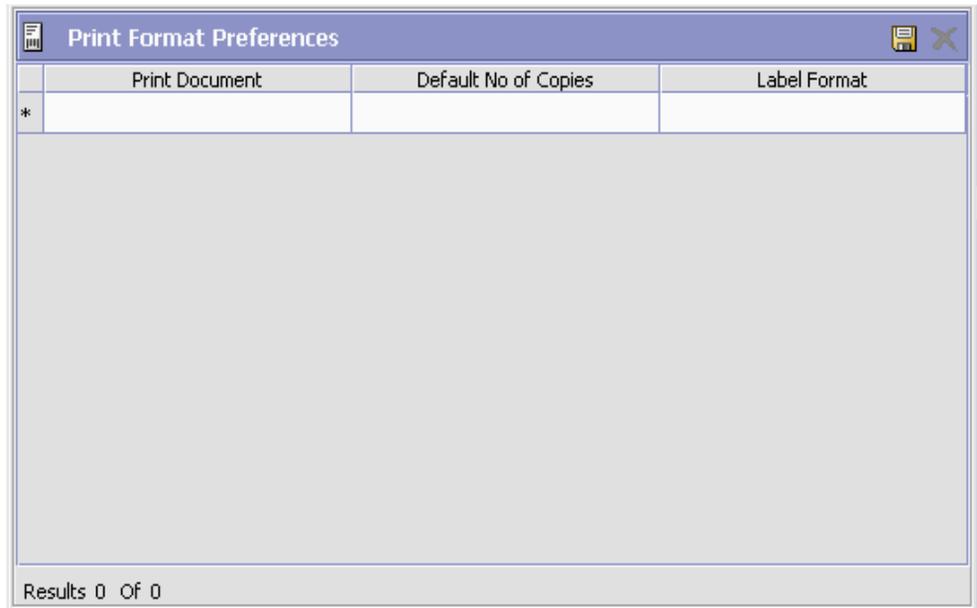


Table 3–5 Seller Attributes Print Format Preferences Tab

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

3.1.2.2.3 Defining an Enterprise’s Inventory Monitoring Rules

If an item is tracked at an Enterprise’s node, you can set up minimum and maximum inventory levels using inventory monitoring rules for an item. When the item reaches one of these levels, a corresponding event is raised if an action is configured. That event can then be configured to trigger an alert, if applicable.

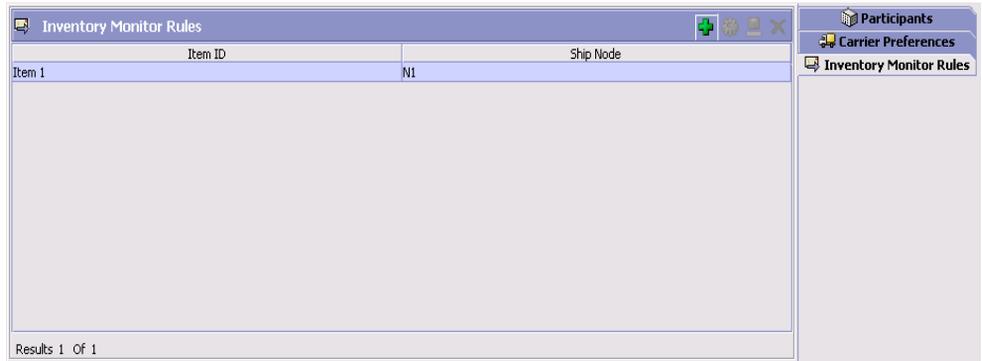
For example, you can set a minimum quantity of 75 for a particular item. When the inventory starts to drop and reaches the minimum level of 75, you can configure the minimum action to raise the Send E-Mail action. This sends an e-mail message to notify a manager of the decreasing inventory level.)

You can add, modify, and delete inventory monitoring rules.

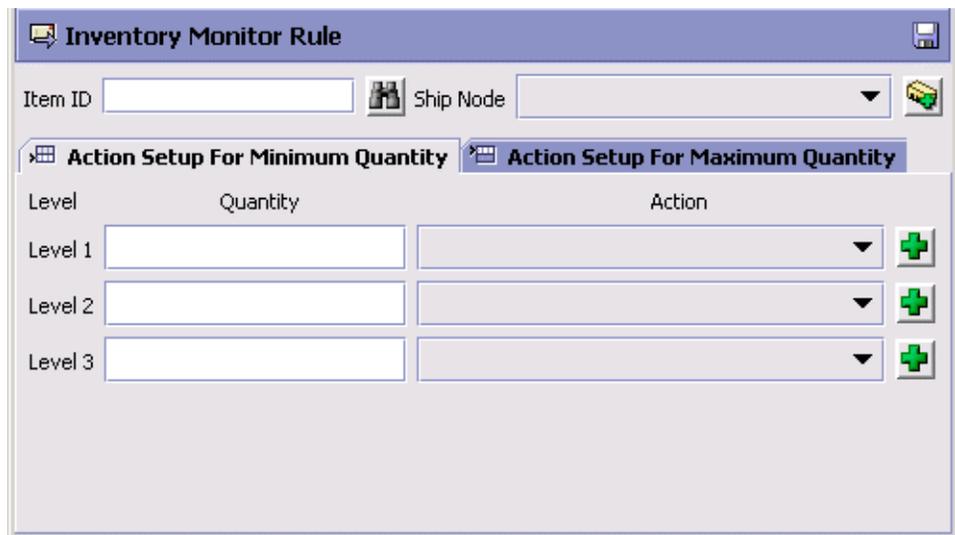
To add an inventory monitoring rule to an Enterprise:

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

2. Choose the Inventory Monitor Rules tab. The Inventory Monitor Rules list displays.



3. Choose . The Inventory Monitor Rule pop-up window displays.
4. Enter information in the applicable fields. Refer to for field value descriptions.
5. Choose .



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](#)

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 .

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 mark for Node ID	Check this box if separate shipments must be created based on the buyer mark for node id.
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.
Level of Service	Check this box if separate shipments must be created based on the order’s level of service.

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.
<p>Do not mix in a load</p> <p>If any of the following attributes are unselected, a shipment can be consolidated to a load based on the attribute.</p> <p>For example, if Addr Line 1 and Name are unselected, shipments that have different first address line but the same address line 2 and 6 can be included in the same load.</p>	
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"](#). 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 displays. For more information about modifying the region schema details, see Section 13.3, "Defining Region Schemas" .
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 Selling and Fulfillment Foundation	<p>Select this to use the Routing Guides maintained in Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation 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 displays 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 displays 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 displays 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, the system re-routes the shipment, 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 displays.
3. Select . A Routing Guide Line Details screen displays 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" .	
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.

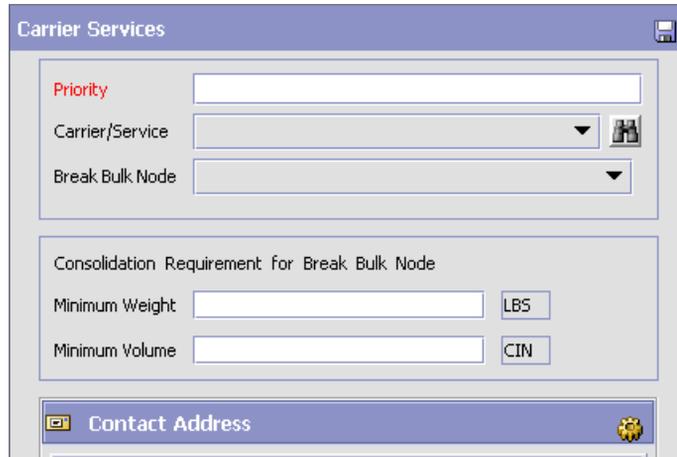
You can use the Carrier Services panel for:

- [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 displays.



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 displays.
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 displays.
2. Select a routing guide line in the Routing Guide Line list window, and select . The Routing Guide Line Details window displays.
3. Enter information in the applicable fields. Refer to [Table 3–10](#) for field value descriptions.
4. Choose .

Deleting a Routing Guide Line

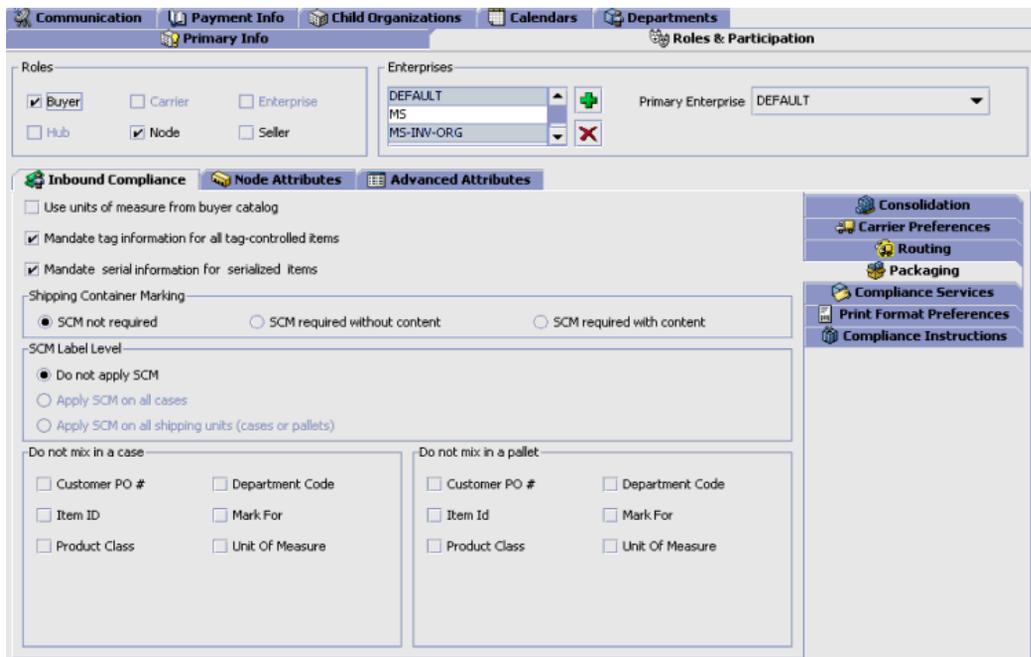
To delete a Routing Guide Line:

1. From the Routing Guide Lines Details window, select the Routing Details Tab. A Routing Guide Line search window displays.
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 .



The screenshot shows the 'Roles & Participation' tab in the 'Organization Details' window. The 'Inbound Compliance' section is active, and the 'Packaging' sub-tab is selected. The 'Roles' section shows 'Buyer' and 'Node' checked. The 'Enterprises' list includes 'DEFAULT', 'MS', and 'MS-INV-ORG', with 'DEFAULT' as the primary enterprise. The 'Inbound Compliance' section includes checkboxes for 'Mandate tag information for all tag-controlled items' and 'Mandate serial information for serialized items'. The 'Shipping Container Marking' section has 'SCM not required' selected. The 'SCM Label Level' section has 'Do not apply SCM' selected. The 'Do not mix in a case' and 'Do not mix in a pallet' sections have no checkboxes selected. A sidebar on the right contains buttons for 'Consolidation', 'Carrier Preferences', 'Routing', 'Packaging', 'Compliance Services', 'Print Format Preferences', and 'Compliance Instructions'.

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.
Mandate Serial information for serialized items	The buyer can mandate vendors to provide serial information for all the serialized items shipped to them. Select this field to ensure that the serial information is provided whenever you ship serialized items to this buyer.
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	

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 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	
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](#)

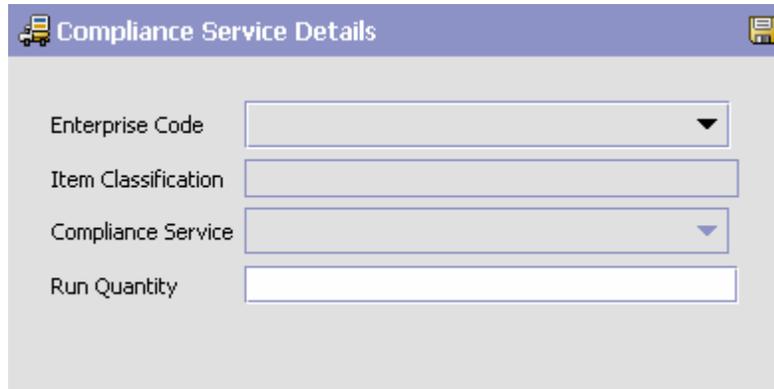
To set up a Buyer's Inbound Compliance Services parameters:

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 displays, enter information in the applicable fields. Refer to [Table 3-13](#) for field value descriptions.
3. Choose .



Compliance Service Details

Enterprise Code

Item Classification

Compliance Service

Run Quantity

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 run. 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 have the compliance services applied. The result is the buyer receives 8 items, 1 from current inventory, 7 that is newly created, and there are 3 newly created items now available in inventory.</p> <p>If the buyer requires more than the run quantity produces, 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 displays.
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 .

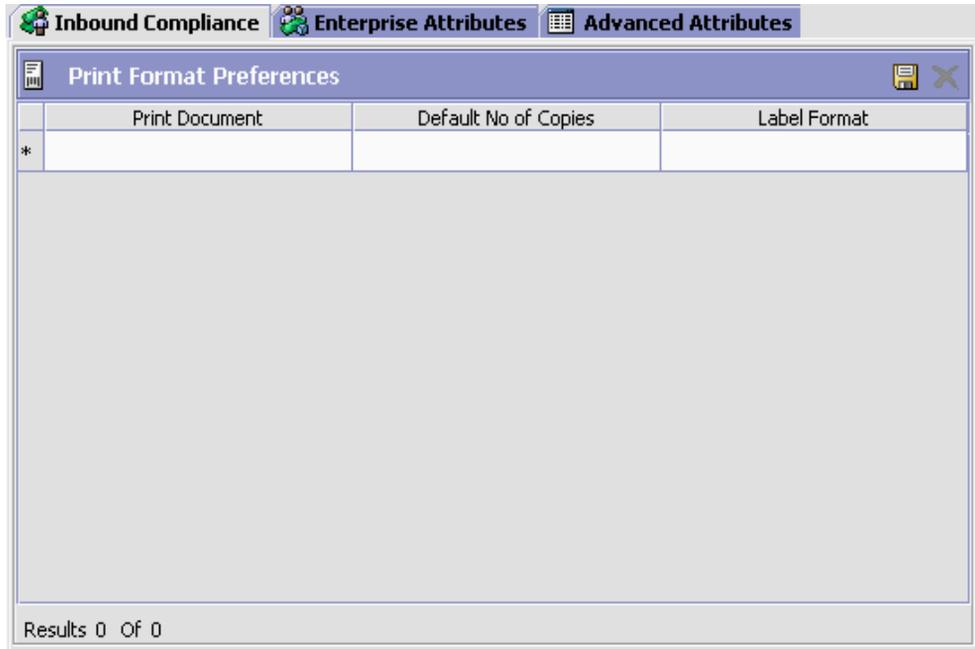


Table 3–14 Inbound Compliance Print Format Preferences Tab

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.

3.1.2.4 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](#)

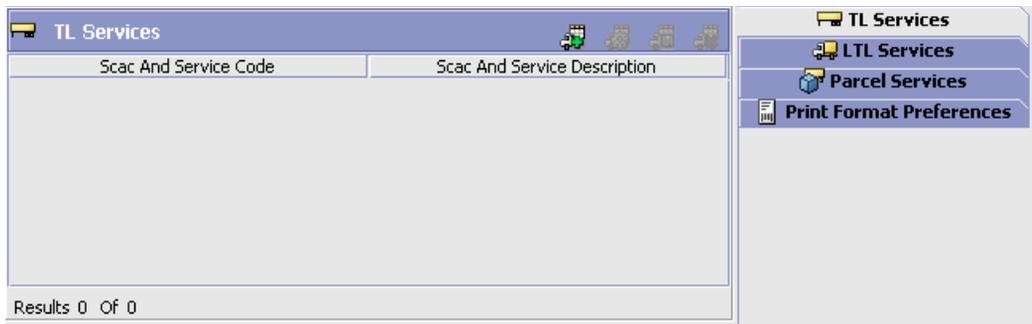
- [Defining Carrier Services for Less-Than Truckload Shipments](#)
- [Defining Carrier Services for Parcel Shipments](#)

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:

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. Choose . The Truckload Service Details pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–15](#) for field value descriptions.
5. Choose .

Figure 3–1 Truckload Service Details

Table 3–15 Truckload Service Details Pop-Up Window

Field	Description
Service	Select the appropriate service from the drop-down list.
Scac and Service	Enter the name of the SCAC and service. Note: This is populated when the carrier service code is chosen. You can then edit it as necessary.
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. You can then edit it as necessary.

Table 3–15 Truckload Service Details Pop-Up Window

Field	Description
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 Selling and Fulfillment Foundation.
Extended Ship Mode	If the carrier service is not TL, LTL, or parcel, select the correct shipment mode from the drop-down list.
Fixed Transit Days	Enter the maximum number of days that the service allows for delivery. For example, 1 Day Air would have a maximum of 1 transit day, whereas Ground may have a maximum of 5 transit days. 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>Sterling Distributed Order Management: Configuration Guide</i> .
Distance Per Day	Enter the maximum distance that the service travels each transit day. Choose the relevant UOM for the distance from the drop-down list. 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>Sterling Distributed Order Management: Configuration Guide</i> .
Allow Shipping Of Hazardous Materials	Check this box to indicate that this carrier allows the 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.
Restrict Shipping to Address of Type	Choose the type of address from the drop-down list where the carrier can ship. If you do not select an options, the carrier can ship to both commercial and residential addresses.

Table 3–15 Truckload Service Details Pop-Up Window

Field	Description
Package Level Integration	Choose this if this carrier service caters to package-level integration.
Shipment Level Integration	Choose this if this carrier service caters to shipment-level integration.
Supports Integration for Return Shipping Label	Check this box to indicate that this carrier service supports integration for return shipping label.

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 displays.
4. Modify information in the applicable fields. Refer to [Table 3–15](#) 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.

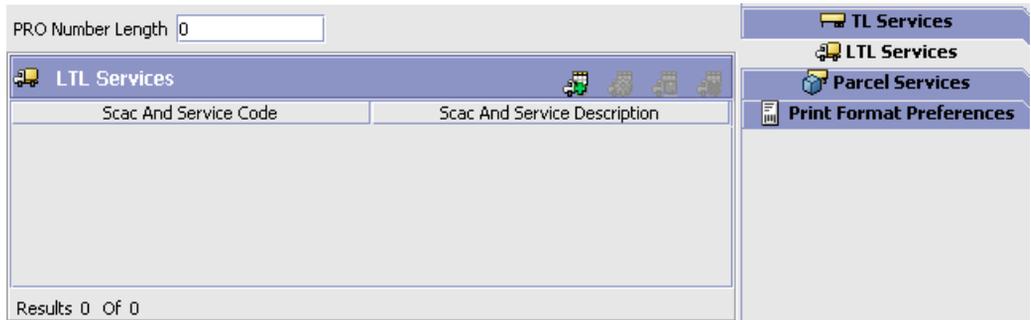
To create carrier services for less-than truckload shipments:

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 3.1.2.5.6, "Defining a Node's LTL Carrier Preferences"](#).

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



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

Figure 3–2 Less-Than Truckload Service Details

Table 3–16 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. You can then edit it as necessary.
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. You can then edit it as necessary.
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 3–16 Less-Than Truckload Service Details Pop-Up Window

Field	Description
Host Code	This field is not used in this version of Selling and Fulfillment Foundation.
Extended Ship Mode	If the carrier service is not TL, LTL, or parcel, select the correct shipment mode from the drop-down list.
Fixed Transit Days	<p>Enter the maximum number of days that the service allows for delivery. For example, 1 Day Air would have a maximum of 1 transit day, whereas Ground may have a maximum of 5 transit days.</p> <p>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>Sterling 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 from the drop-down list.</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>Sterling Distributed Order Management: Configuration Guide</i>.</p>
Allow Shipping Of Hazardous Materials	Check this box to indicate that this carrier allows the 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.
Restrict Shipping to Address of Type	<p>Choose the type address from the drop-down list where the carrier can ship.</p> <p>If you do not select an option, the carrier can ship to both commercial and residential addresses.</p>
Package Level Integration	Select this option if this carrier service caters to package-level integration.

Table 3–16 Less-Than Truckload Service Details Pop-Up Window

Field	Description
Shipment Level Integration	Select this option if this carrier service caters to shipment-level integration.
Supports Integration for Return Shipping Label	Check this box to indicate that this carrier service supports integration for return shipping label.

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 3.1.2.5.6, "Defining a Node's LTL Carrier Preferences"](#).

4. Select the applicable carrier service and choose . The Less-Than Truckload Service Details pop-up window displays.
5. Modify information in the applicable fields. Refer to [Table 3–16](#) 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

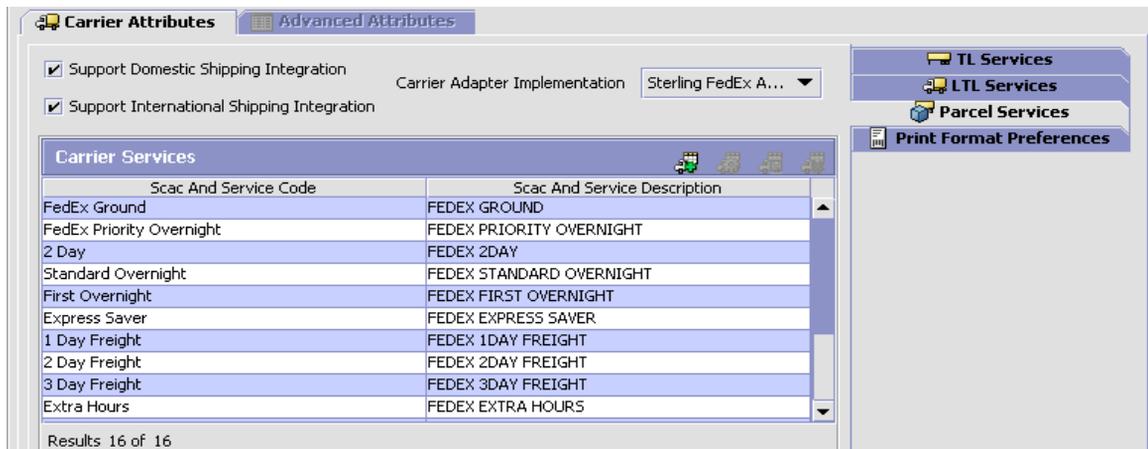
A parcel is the smallest shipping mode. A shipment is normally referred to as a parcel when it weighs under 150 pounds. You can add, modify, and delete carrier services for the parcel shipping mode.

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. Choose the Carrier Adapter that you want to integrate with the carrier from the Carrier Adapter Implementation drop-down list.

Note: Only the Carrier Adapters supported for the selected carrier will be displayed in the drop-down list.

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



6. Choose . The Parcel Service Details pop-up window displays.
7. Enter information in the applicable fields. Refer to [Table 3–17](#) for field value descriptions.
8. Choose .

Figure 3–3 Parcel Service Details Pop-Up Window

Table 3–17 Parcel Service Details 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. You can then edit it as necessary.
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. You can then edit it as necessary.

Table 3–17 Parcel Service Details Pop-Up Window

Field	Description
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 Selling and Fulfillment Foundation.
Extended Ship Mode	If the carrier service is not TL, LTL, or parcel, select the correct shipment mode from the drop-down list.
Fixed Transit Days	Enter the maximum number of days that the service allows for delivery. For example, 1 Day Air would have a maximum of 1 transit day, whereas Ground may have a maximum of 5 transit days. 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>Sterling Distributed Order Management: Configuration Guide</i> .
Distance Per Day	Enter the maximum distance that the service travels each transit day. Choose the relevant UOM for the distance from the drop-down list. 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>Sterling Distributed Order Management: Configuration Guide</i> .
Allow Shipping Of Hazardous Materials	Check this box to indicate that this carrier allows the 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.
Restrict Shipping to Address of Type	Choose the type of address from the drop-down list where the carrier can ship. If you do not select an option, the carrier can ship to both commercial and residential addresses.

Table 3–17 Parcel Service Details Pop-Up Window

Field	Description
Package Level Integration	Choose this option if this carrier service caters to package level integration.
Shipment Level Integration	Choose this option if this carrier service caters to shipment level integration.
Supports Integration for Return Shipping Label	Check this box to indicate that this carrier service supports integration for return shipping label.

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 displays.
4. Modify information in the applicable fields. Refer to [Table 3–17](#) 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 3–18](#) for field value descriptions.
4. Choose .

Figure 3–4 Node Primary Info

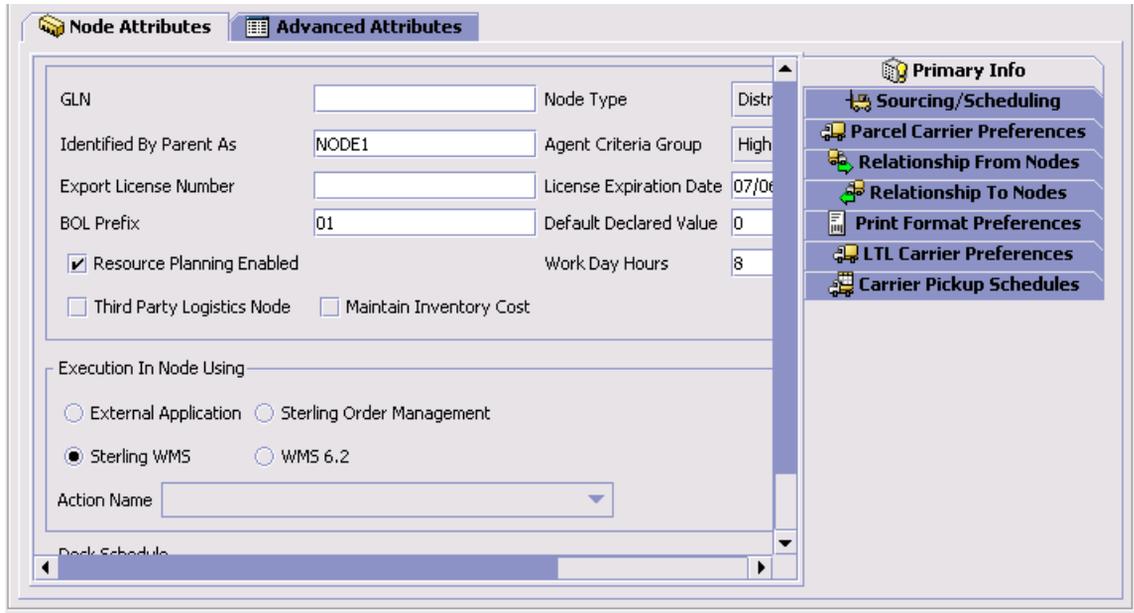


Table 3–18 Node Primary Info Tab

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 3–18 Node Primary Info Tab

Field	Description
Default Declared Value	Enter the price that is to be displayed as the default declared value in the Application Console. This price is typically used by parcel carriers for computing insurance.
Work Day Hours	The standard number of working hours per resource or person in a day. This is used to convert hours to days.
Resource Planning Enabled	Choose this to enable the planning of resources and activities.
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.
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" .
Sterling Order Management	Choose this to have order releases interface through the Application Console.
Sterling WMS	Choose this to have order releases interface through a Sterling WMS.
WMS 6.2	Choose this to have order releases interface through a Sterling WMS version prior to and including Release 6.2.

Table 3–18 Node Primary Info Tab

Field	Description
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".</p> <p>Note: In the drop-down, only the Actions linked to the Primary Enterprise of this Node/Organization are available. Any actions created from this screen using the + button are 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>
Dock Schedule	
Dock Schedules are managed by	Select the application that manages the dock schedule.

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 the node.

To define node sourcing and scheduling:

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

Figure 3–5 Node Attributes

Table 3–19 Node Sourcing/Scheduling Tab

Field	Description
Do not schedule an order to this node more than <number of days> 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 3–19 Node Sourcing/Scheduling Tab

Field	Description
Procure To Ship Allowed	<p>Check this box 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 life cycle 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>Check this box if you want this node to accept a procurement to confirm availability before proceeding with the order.</p>
Work Order Creation Is Allowed	<p>Check this box 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>Check this box if substitution of product items within an order is allowed.</p>

Table 3–19 Node Sourcing/Scheduling Tab

Field	Description
Use End Of Shift Time	<p>Check this box if you want the node to base shipment time by the end of the next feasible shift.</p> <p>Uncheck this box 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 on Friday at 1PM, the order is scheduled to ship on same day at 4PM if Use End Of Shift Time box is checked. The order is scheduled to ship on same day at 3PM if Use End Of Shift Time box is unchecked.</p> <p>If an order is sent to a node on Friday at 3PM, the order scheduled to ship on same day at 8PM if Use End Of Shift Time box is checked. The order is scheduled to ship on same day at 5PM if Use End Of Shift Time box is unchecked.</p> <p>Note: Use End Of Shift Time is only applicable to nodes that use a shipping calendar that has shift times defined.</p> <p>Note: Use End Of Shift Time is only applicable for product lines.</p>
Gift Wrap Services Allowed	<p>Check this box to indicate that this node provides gift wrapping services.</p>
Item Based Allocation Allowed	<p>Check this box to allow item based allocation for the item. When the 'Use Item Based Allocation' rule is enabled, the item based allocation are only applicable for the items and nodes which have the Item Based Allocation Allowed attribute enabled. For more information about item-based allocation, see the <i>Selling and Fulfillment Foundation: Product Concepts Guide</i>.</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>

Table 3–19 Node Sourcing/Scheduling Tab

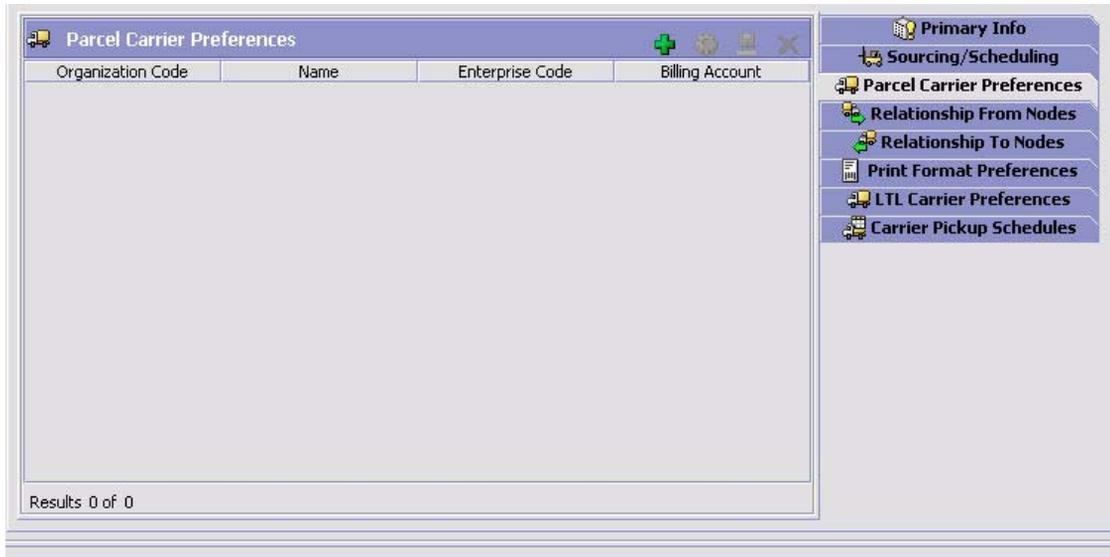
Field	Description
Shipping Calendar	Select the calendar to use to determine the available shifts for shipping from the node. The calendars of the node as well as the calendars of the primary enterprise of the node display in this drop-down list.
Receipt Processing Time For Procurement (Hours)	Enter how many hours it takes the node to process receipts for procurement.
Receipt Processing Time for Forwarding (Hours)	Enter the time it takes the node to process receipts for forwarding in hours.
Receipt Processing Time for Future Inventory (Days)	Enter how many days are required to process incoming future supplies before they are available for orders.
Node Can Ship To	
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.	
Any Address	Check this box to allow this node to ship to any address.
All Nodes	Select this option if this node can ship to all nodes.
Specific Nodes Of Type	If this node can only ship to nodes with a specific node type, select this option, and check the applicable node types available.

3.1.2.5.3 Defining a Node’s Parcel Carrier Preferences

You can identify the carriers that 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 3–20](#) for field value descriptions.
5. Choose .

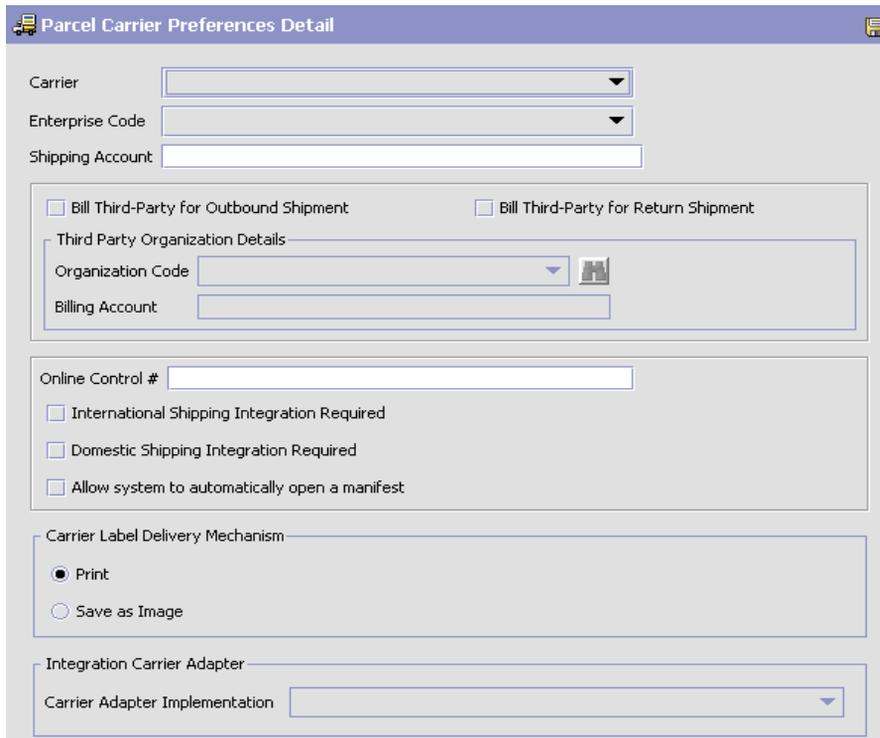


Table 3–20 Parcel Carrier Preferences Detail Pop-up Window

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.
Bill Third Party for Outbound Shipment	Check this box to bill shipping charges to a third-party organization.
Bill Third Party for Return Shipment	Check this box to bill return shipping charges 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.
Carrier Label Delivery Mechanism	
Print	Choose this option to print the carrier label automatically.
Save as Image	Choose this option to save the carrier label in PNG format.
Integration Carrier Adapter	
Carrier Adapter Implementation	Choose the Carrier Adapter that you want to integrate with the selected carrier.

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 3–20](#) 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. For more information on creating Transfer Schedules, refer to section [Section 3.1.2.5.5, "Defining a Transfer Schedule"](#).

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

Table 3–21 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 are sent. For the Relationship To Node tab, this option is defaulted to the node you are configuring and disabled.
To Node	Select the node at which transfer order items are received. For the Relationship From Node tab, this option is defaulted to the node you are configuring and disabled.
Transfer Schedules	
Effective From	Indicates the date on which the schedule takes effect.
Effective To	Indicates the date on which the specified transfer schedule stops being effective.
Days of the Week	Indicates on which days during the transfer schedule items are eligible for items to ship.

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 displays.
4. Enter information into the applicable fields. Refer to [Table 3–21](#) 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 Transfer Schedule

Transfer schedules provide a means to allow or disallow a transfer during a specific period of time.

Creating a Transfer Schedule

To create a transfer schedule:

1. From the Relationship Details screen, choose  from the Transfer Schedules panel. The Transfer Schedule window displays.
2. Enter information into the applicable fields. For field value descriptions see [Table 3–22](#).
3. Click .

 **Transfer Schedule** 

Effective From Date  Effective To Date 

Default Transit Days

Shipped On

Day of Week	Override Transit Days	Day of Week	Override Transit Days
<input type="checkbox"/> Sunday	<input type="text"/>	<input type="checkbox"/> Monday	<input type="text"/>
<input type="checkbox"/> Tuesday	<input type="text"/>	<input type="checkbox"/> Wednesday	<input type="text"/>
<input type="checkbox"/> Thursday	<input type="text"/>	<input type="checkbox"/> Friday	<input type="text"/>
<input type="checkbox"/> Saturday	<input type="text"/>		

 **Ship Date Overrides** 

	Ship Date	Can Ship	Override Transit Days
*	<input type="text"/>	<input type="text"/>	<input type="text"/>

Results 0 of 0

Table 3–22 Transfer Schedule Window

Field	Description
Effective From Date	Select the date on which the transfer schedule becomes effective. If this value is left empty, it is assumed that the Transfer Schedule has been effective for an infinite number of days.
Effective To Date	Select the date on which the transfer schedule ends. If this value is left empty, it is assumed that the Transfer Schedule remains effective indefinitely.
Default Transit Days	Enter the minimum number of days shipments are expected to take to reach the end location.
Shipped On	
Day Of Week	Check the box for each day of the week on which transfers are permitted.
Override Transit Days	Enter the number of transit days for a specific day of the week. For example, shipping on a Saturday may add one day to the number of transfer days. As a result, the new Transfer Days value would be 1 + the amount of the standard value.
Ship Date Overrides	
Ship Date	Select the date for which you would like to override shipping. For example, Tuesday, December 25, 2007, a holiday.
Can Ship	Select Yes to allow or No to disallow shipping on the specified day.
Override Transit Days	Enter the override transit days for the override dates.

Note: If a transfer schedule exists for one day, it is assumed that a transfer schedule exists for all days.

Modifying a Transfer Schedule

To modify a transfer schedule:

1. From the Relationship Details screen, Transfer Schedules panel, select the Transfer Schedule you would like to modify.
2. Choose .
3. Enter information into the applicable fields. For field value descriptions, see [Table 3–22](#).
4. Click .

Deleting a Transfer Schedule

To delete a transfer schedule:

1. From the Relationship Details screen, Transfer Schedules panel, select the Transfer Schedule you would like to delete.
2. Choose .

3.1.2.5.6 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.

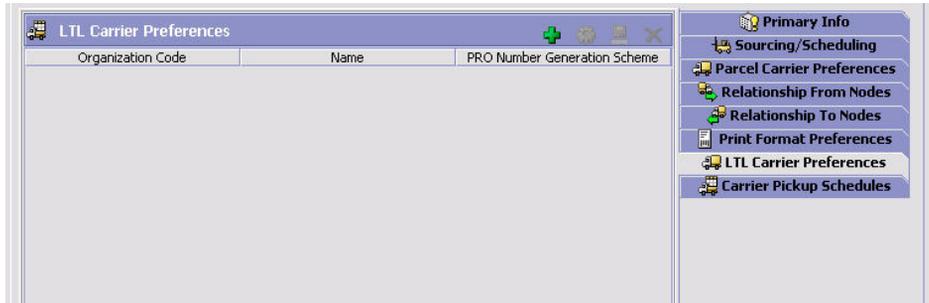
A warehouse may define the range of PRO Numbers assigned by a carrier. The PRO Number has a fixed length defined for each carrier, and may contain a prefix.

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" pop-up window. It has a title bar with a truck icon and a save icon. The main content area is divided into several sections:

- PRO Number Generation Scheme**: A dropdown menu for "Carrier" and three radio buttons: "System Generated" (selected), "Externally Generated", and "Do Not Generate".
- Set up data for System Generated scheme**: A section containing two sub-sections:
 - PRO Number Set - 1**: Three text input fields for "PRO Number Prefix", "PRO Number Start", and "PRO Number End".
 - PRO Number Set - 2**: Three text input fields for "PRO Number Prefix", "PRO Number Start", and "PRO Number End".
- Notification**: A text input field for "Notification Threshold" and a dropdown menu for "Notification Service".
- Check Digit Determination**: A text input field for "Select Algorithm" 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"> • Sterling WMS generates the PRO Numbers • Sterling WMS uses the PRO Numbers generated by an external system, or • Sterling WMS does not generate PRO Numbers.
Carrier	Choose the carrier whose PRO Number Generation Scheme is being defined.
System Generated	Select 'System Generated' if Sterling WMS should generate PRO Numbers.
Externally Generated	Select 'Externally Generated' if Sterling 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 Sterling WMS generates the PRO Numbers.
PRO Number Sets	When the first range of PRO Numbers is used up, Sterling 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 Sterling 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>
PRO Number Set - 2	Defines the second range of PRO Numbers to be used by Sterling WMS.

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

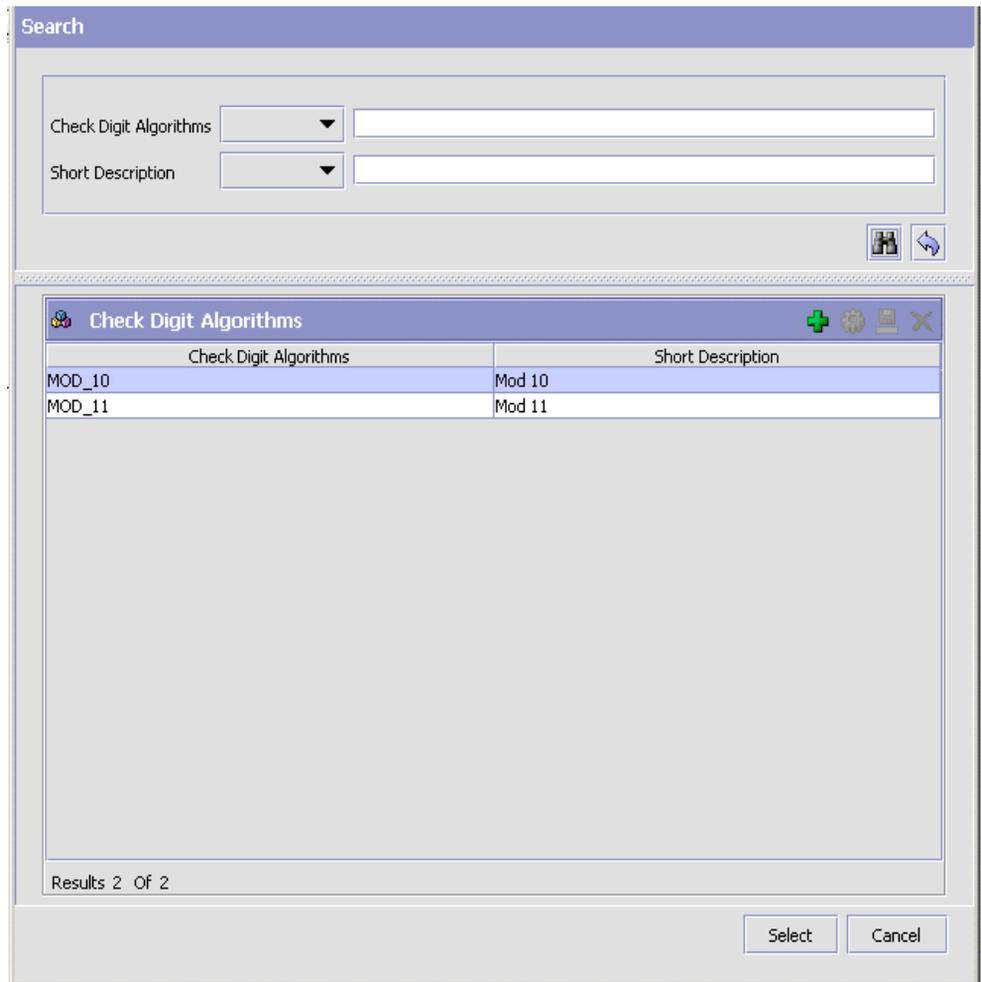
Field	Description
PRO Number Prefix	Enter the PRO Number Prefix to be used in generating the second range of PRO Numbers. PRO Number Prefix can be alpha numeric, and is not included for computing the check digit.
PRO Number Start	Enter the PRO Number Start to be used in generating the second range of 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, Sterling 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 is invoked when Notification Threshold is reached.
Check Digit Determination	Defines the check digit algorithm to be used for PRO Generation.
Select Algorithm	Choose the relevant algorithm for check digit determination. Typical values are mod-11 and mod-10. Note: When 'mod-11' or 'mod-10' schema of check digit determination is chosen, the check digit is generated out-of-the-box by Sterling WMS. For more details about Check Digit Determination Algorithm, see Setting Up a Check Digit Determination Algorithm .

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 displays.



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

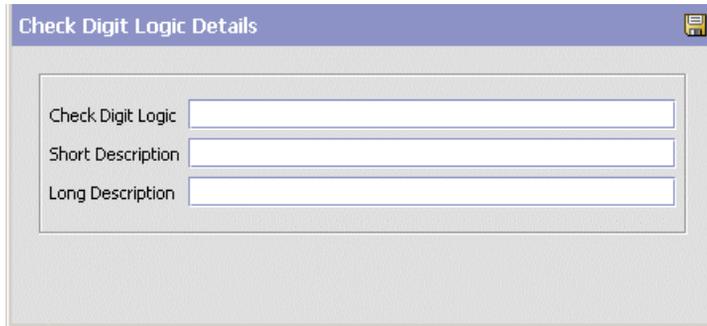


Table 3–24 *Check Digit Logic Details Popup Window*

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 displays.
3. In the Search panel, enter applicable search criteria, and choose .

4. The relevant search results display 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 displays.
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 displays.
3. In the Search panel, enter applicable search criteria, and choose .
4. The relevant search results display 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 displays.
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 displays.
3. In the Search panel, enter applicable search criteria, and choose .
4. The relevant search results display 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 .

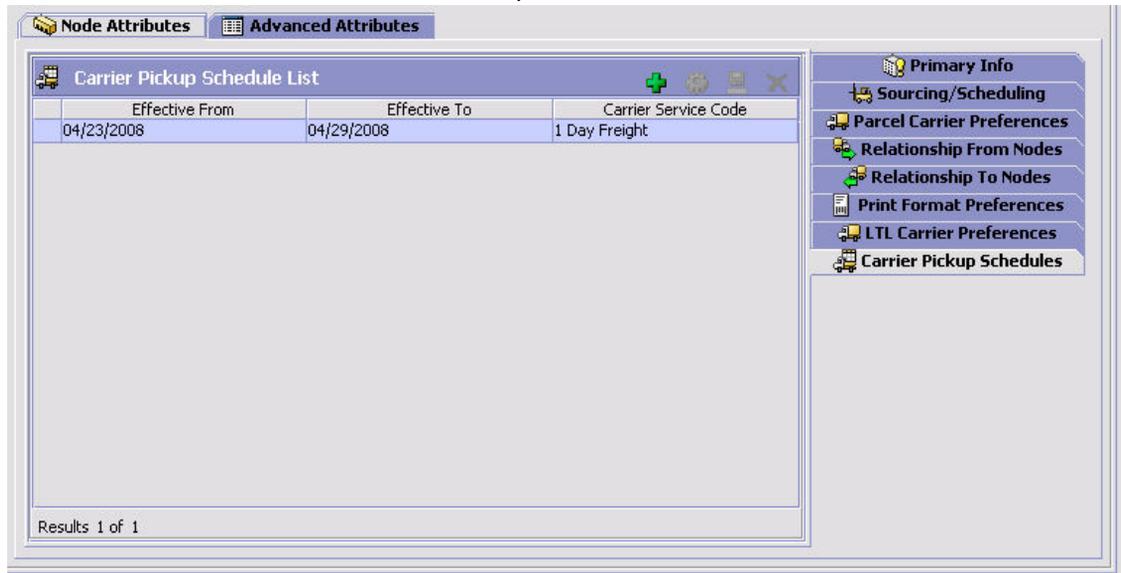
3.1.2.5.7 Defining a Node's Carrier Pickup Schedule

You can define a carrier's pickup schedule at a node. You can add, modify, and delete pickup schedules. See the *Selling and Fulfillment Foundation: Product Concepts Guide* for more information.

To add a carrier pickup schedule to a node:

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

2. Choose the Carrier Pickup Schedules tab.



3. Choose . The Carrier Pickup Schedule Details pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–25](#) for field value descriptions.

5. Choose .

Table 3–25 *Carrier Pickup Schedule Details Popup Window*

Field	Description
Carrier Service	Select the carrier service for which you are configuring a pickup schedule.
Effective From Date	Select the date on which the pickup schedule becomes effective. If you do not specify a start date, the system assumes that the pickup schedule has already started.
Effective To Date	Select the date on which the pickup schedule ends. If you do not specify an end date, the pickup schedule is in effect forever. Ensure that you specify a date range that is not already in use by another carrier pickup schedule. For example, if a carrier pickup schedule has a date range of January through June, you cannot specify a date range of May through July for a different schedule.

Field	Description
Pick Up On	
Day of Week	Check the box for each day of the week on which pickups are permitted.
Pickup Time	Enter the time of day when this carrier service picks up shipments from this node. If you do not specify a time, the carrier does not pick up on the selected day.
Pickup Date Overrides	
Override Date	Select the date for which you want to override the scheduled pickup options.
Can Pick Up	Select Yes to allow pick up or No to disallow pick up on the specified date.
Pickup Time	Enter the pickup time for the override date that you are allowing pickup.

Modifying a Carrier Pickup Schedule

To modify a carrier pickup schedule:

1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the Carrier Pickup Schedules tab.
3. Select a pickup schedule from the Carrier Pickup Schedule List and choose . The Carrier Pickup Schedule Details popup window displays.
4. Enter information in the applicable field. Refer to [Table 3–25](#) for field value descriptions.
5. Choose .

Deleting a Carrier Pickup Schedule

To delete a carrier pickup schedule:

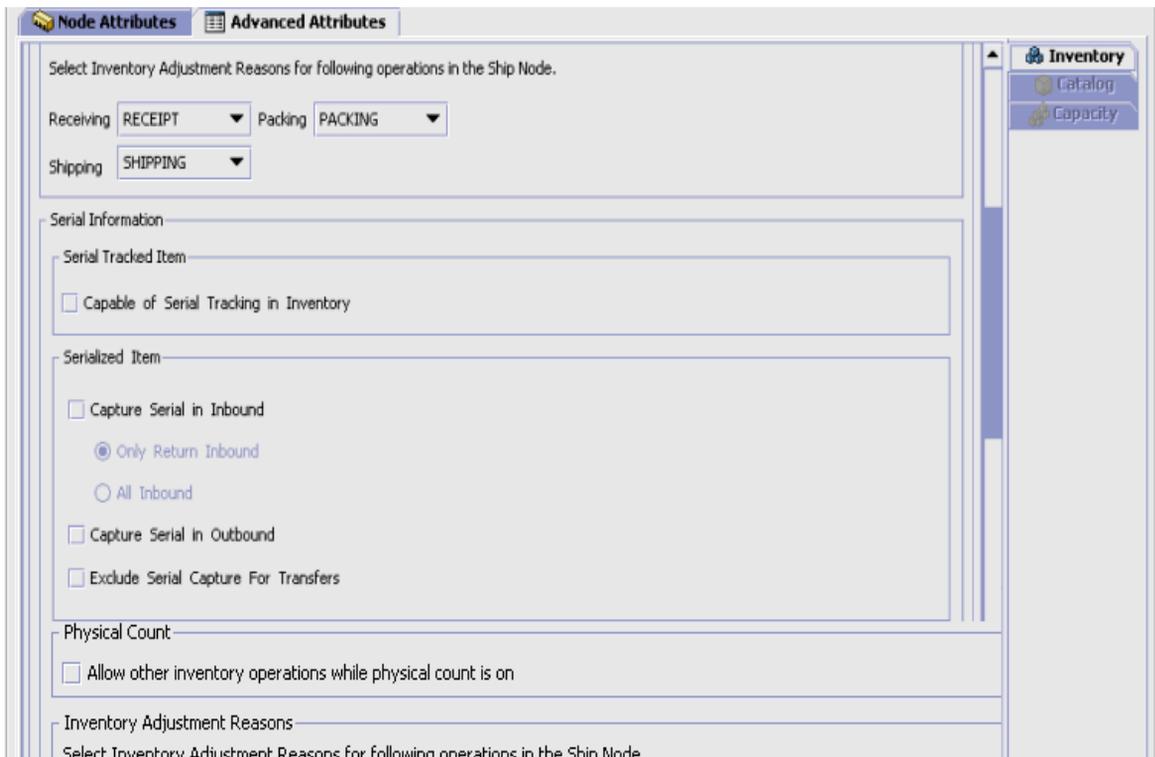
1. From the Roles & Participation tab in the Organization Details window, choose Node Attributes.
2. Choose the Carrier Pickup Schedules tab.
3. Select a pickup schedule from the Carrier Pickup Schedule List and choose .

3.1.2.6 Defining a Node's Advanced Inventory Attributes

You can determine if a node's inventory is maintained within Selling and Fulfillment Foundation. 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 3–26](#) for field value descriptions.
3. Choose .



The screenshot shows the 'Node Attributes' window with the 'Advanced Attributes' tab selected. The main content area is titled 'Select Inventory Adjustment Reasons for following operations in the Ship Node.' and contains three dropdown menus: 'Receiving' set to 'RECEIPT', 'Packing' set to 'PACKING', and 'Shipping' set to 'SHIPPING'. Below these are sections for 'Serial Information' and 'Physical Count'. The 'Serial Information' section includes 'Serial Tracked Item' with a checkbox for 'Capable of Serial Tracking in Inventory', and 'Serialized Item' with checkboxes for 'Capture Serial in Inbound', 'Capture Serial in Outbound', and 'Exclude Serial Capture For Transfers'. Under 'Capture Serial in Inbound', there are radio buttons for 'Only Return Inbound' (selected) and 'All Inbound'. The 'Physical Count' section has a checkbox for 'Allow other inventory operations while physical count is on'. The 'Inventory Adjustment Reasons' section at the bottom is partially visible and repeats the instruction to select reasons for operations in the Ship Node. On the right side, there is a vertical sidebar with an 'Inventory' header and two sub-items: 'Catalog' and 'Capacity'.

Table 3–26 Advanced Attributes, Inventory Tab

Field	Description
Inventory Information Not Available To Selling and Fulfillment Foundation	<p>Select this option if the node uses an inventory application other than Selling and Fulfillment Foundation to track inventory.</p> <p>If this option is selected, Selling and Fulfillment Foundation never has access to the inventory picture for this node, and never attempts to do any inventory availability checks.</p>
Inventory Is Made Available To Selling and Fulfillment Foundation	<p>Select this if the node tracks inventory using either Selling and Fulfillment Foundation or another inventory application. The node's inventory picture can be used in Selling and Fulfillment Foundation 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>
Get External Supply Real Time	<p>Select this field if the supply is available to Selling and Fulfillment Foundation, but comes from an external system instead of being maintained internally. The supply picture is obtained by calling the <code>getExternalSupply</code> user exit.</p> <p>Note: This checkbox is only available when the inventory information is available to Selling and Fulfillment Foundation, and the organization is a node only.</p>
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.
Serial Information	
Serial Tracked Item	

Table 3–26 Advanced Attributes, Inventory Tab

Field	Description
Capable of Serial Tracking in Inventory	Select this to track serials for serial tracked items in inventory. If this is not selected, serial tracking will be based on configuration for serialized items.
Serialized Item	
Capture Serial in Inbound	Select this to capture serial information in inbound. There are two scenarios of capturing serials: Only Return Inbound - Select this to capture only for return inbound items. All Inbound - Select this to capture serials for all inbound inventory.
Capture Serial in Outbound	Select this to capture the serial information when shipping.
Exclude Serial Capture For Transfer	Select this to exclude serial capturing for transfer orders.
Physical Count	
Allow Other Inventory Operations While Physical Count Is On	Check this box to allow other inventory operations to be performed during the physical count process.
Inventory Adjustment Reasons	
<p>You can associate inventory adjustment reasons for a ship node that performs receiving, packing, and shipping activities.</p> <p>Based on the configurations done for a ship node in the Sterling Warehouse Management System, the inventory adjustment reasons drop-down list displays. For more information about defining inventory adjustment reasons, see the <i>Sterling Warehouse Management System: Configuration Guide</i>.</p>	
Receiving	Select the adjustment reason code associated with the receiving activity from the drop-down list. By default, the value of the receiving activity is RECEIPT. Note: Ensure that the reason code is not associated with an accounting bin.

Table 3–26 Advanced Attributes, Inventory Tab

Field	Description
Packing	Select the adjustment reason code associated with the packing activity from the drop-down list. By default, the value of the packing activity is PACK. Note: Ensure that the reason code is associated with an accounting bin.
Shipping	Select the adjustment reason code associated with the shipping activity from the drop-down list. By default, the value of the shipping activity is SHIP. Note: Ensure that the reason code is not associated with an accounting bin.
OverPick For Voice Based Tasks	Select the adjustment reason code associated with the voice-based overpicking activity from the drop-down list. Note: This reason code is applicable only when voice-based picking is implemented in a warehouse.

3.1.2.7 Defining an Organization’s Advanced Inventory Attributes

You can determine if an organization’s inventory is maintained within Selling and Fulfillment Foundation. 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 is the inventory organization of the source organization. If the source organization is its own inventory organization, the source organization is 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 .



The screenshot shows the 'Advanced Attributes' tab in the Organization Details window. The 'Inventory Information' section is active, with 'Inventory Information Is Available' selected. Under this, 'This Organization Is An Inventory Organization' is selected, and 'Get External Inventory Real Time' is unchecked. The 'Inventory Organization Is' dropdown is empty. Below this is the 'Inventory Organization Relationship' table.

Consumable Inventory Organization	Organization Name
E1[C1]	E1[C1]
ENT-01[C-01]	Enterprise 01 Colony 01

Results 2 of 2

Table 3–27 Inventory Tab

Field	Description
Inventory Information Is Not Available	<p>Select this option if the organization uses an inventory application other than Selling and Fulfillment Foundation to track supply and demand.</p> <p>If this option is selected, Selling and Fulfillment Foundation does not have access to the inventory picture for this organization, and do not attempt to do any inventory availability checks. Orders can still be placed using this organization as a seller, but the orders are scheduled without any inventory availability checks.</p>
Inventory Information Is Available	<p>Select this if the organization tracks inventory supply and demand using either Selling and Fulfillment Foundation or another inventory application. The organization's supply and demand picture can be used in Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation 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 3–27 Inventory Tab

Field	Description
Inventory Organization Is	<p>Select this and the applicable inventory organization if you selected Inventory Is Made Available To Selling and Fulfillment Foundation 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 External Inventory Real Time	<p>Select this field if the inventory is available to Selling and Fulfillment Foundation, but comes from an external system instead of being maintained internally. The inventory picture is then 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 Selling and Fulfillment Foundation, but comes from an external system instead of being maintained internally. The supply picture is then 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>
Physical Count	
Allow Other Inventory Operations While Physical Count Is On	<p>Check this box to allow other inventory operations to be performed during the physical count process.</p> <p>This checkbox is enabled only if the organization is a node.</p>
Inventory Organization Relationship	
Inventory Organization Relationship	<p>Choose  to select an inventory organization from the drop-down list. It displays Consumable Inventory Organization and Organization Name.</p> <p>Note: This panel is visible only if This Organization Is An Inventory Organization is selected.</p>

Note: A consumable inventory organization cannot be a consuming inventory organization of another inventory organization. This means that only one level of relationship is allowed. There is no limit on the number of consumable inventory organizations a consuming inventory organization can have.

Note: Inventory rules are read based on the consuming inventory organization. In a configuration where a consumable inventory organization participates with multiple consuming inventory organizations, the system validates whether all the participating consuming inventory organizations have the same set of inventory rules. These inventory rules are Rules within ATP rules, Use Item Based Allocation, Use Future Supply of Inventory Node Control, Future Supply Type Safety Factor Percentage, Onhand Supply Type Safety Factor Percentage, and Default Distribution Rule ID. For more details about inventory organizations, refer to *Selling and Fulfillment Foundation: Product Concepts Guide*.

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 is the catalog organization of the source organization. If the source organization is its own catalog organization, the source organization is 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 3–28](#) for field value descriptions.
4. Choose .



The screenshot shows the 'Advanced Attributes' tab in the 'Organization Details' window. The 'Catalog' sub-tab is selected. The main area contains two radio buttons: 'Organization Defines Its Own Catalog' (which is selected) and 'Catalog Defined By' (which is unselected). The 'Catalog Defined By' option has a dropdown menu next to it. On the right side, there is a vertical toolbar with three icons: 'Inventory', 'Catalog', and 'Capacity'.

Table 3–28 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 is the capacity organization of the source organization. If the source organization is its own capacity organization, the source organization is 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 3–29](#) for field value descriptions.
4. Choose .

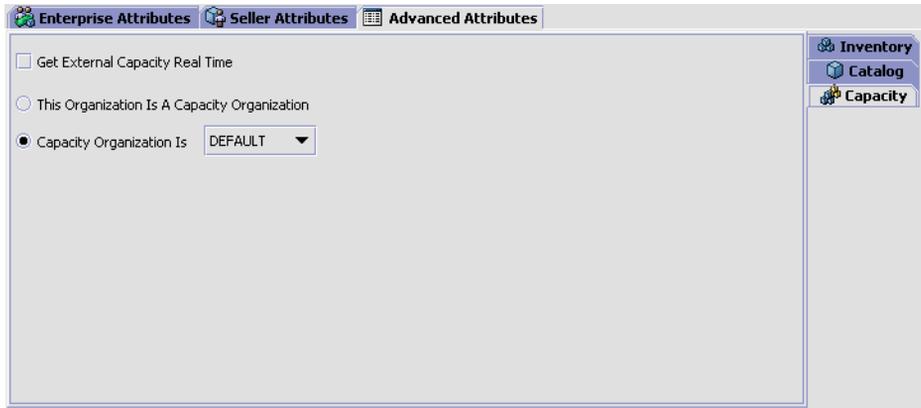


Table 3–29 Capacity Tab

Field	Description
This Organization Is A Capacity Organization	<p>Select This Organization Is A Capacity Organization if you want to specify this organization as a 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 3–29 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 Selling and Fulfillment Foundation from an external system. When Selling and Fulfillment Foundation checks for capacity, the external system provides the capacity in response to a user exit call from Selling and Fulfillment Foundation.

3.1.2.10 Defining an Organization’s Advanced Customer Attributes

You can determine if an organization defines its own customers, or if its customers are defined by another organization.



To define an organization’s advanced customer attributes:

1. From the Roles & Participation tab in the Organization Details window, choose Advanced Attributes.
2. Choose the Customer tab.
3. Enter information in the applicable fields. Refer to [Table 3–29](#) for field value descriptions.
4. Choose .

3.1.2.11 Defining an Organization's Advanced Pricing Attributes

You can determine if an organization maintains its own pricing 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.

To define an organization's advanced pricing attributes:

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

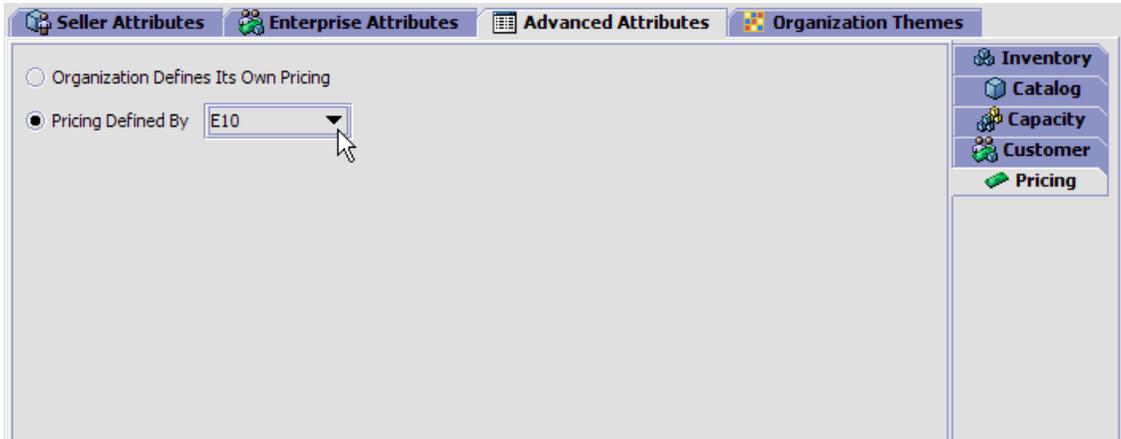
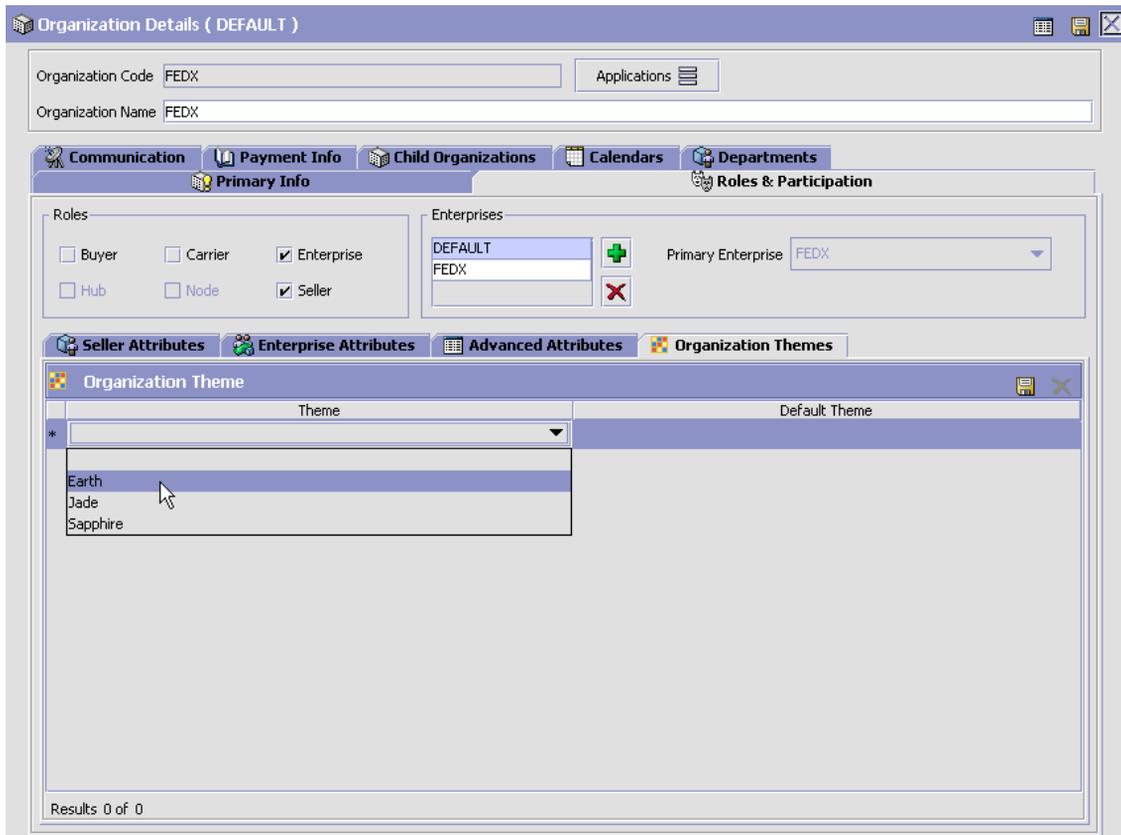


Table 3–30 Pricing Tab

Field	Description
Organization Defines Its Own Pricing	Select this field if the organization defines its own price lists, pricing rules, and coupons. The pricing data that this organization defines can be shared with other organizations.
Pricing Defined By	Select this field and select the applicable organization if you want to use that organization's price lists, pricing rules, and coupons.

3.1.2.12 Defining an Organization's Themes

You can create and maintain a list of themes to be part of your storefront's URL and provide a brand-name, "look-and-feel" entry point for customers on the web. This tab is enabled when an organization has the roles of enterprise and seller. You can list specific themes such as seasonal or promotional displays as well as default themes that are more generic in nature.



To define an Organization's Themes:

1. From the Roles & Participation tab in the Organization Details window, choose Organization Themes.
2. Enter information in the applicable fields. Refer to [Table 3–31](#) for field value descriptions.
3. Choose .

Table 3–31 Organization Theme

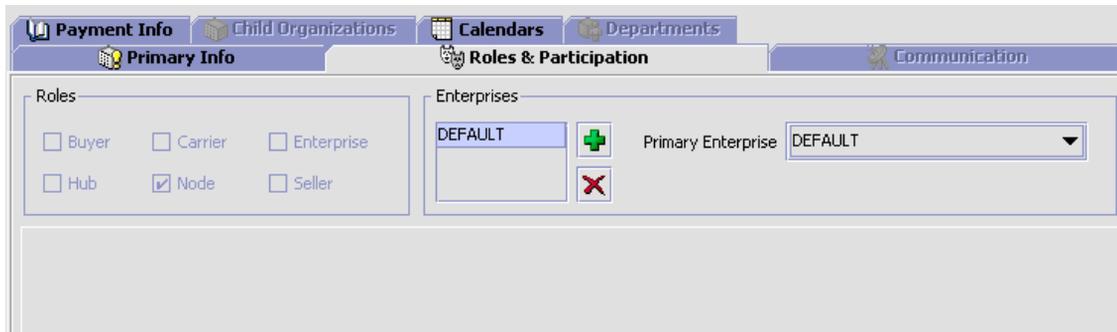
Field	Description
Theme	Enter the name of a theme that your organization supports. This name will become part of the URL.
Default Theme	Enter the ID of your organization's Default Theme. This name will become part of the URL.

3.1.3 Assigning a Yard's Roles and Participant Associations

If an organization is a yard, the role is selected as a Node. You cannot assign any other roles to that yard.

To assign a yard's roles and participant associations:

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



The Roles box is disabled as the organization is a yard and not a fulfillment node.

2. In the Enterprises box, choose **+**. The Participating Enterprises pop-up window is displayed. Select the enterprise that this organization participates with from the drop-down list.
Choose **-** to delete the enterprise that do not participate with the organization.

3. In Primary Enterprise, select the primary Enterprise for the organization from the drop-down list, if applicable.
4. Click .

3.1.4 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](#)

3.1.4.1 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 display.
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 3–32](#) for field value descriptions.
4. Choose .

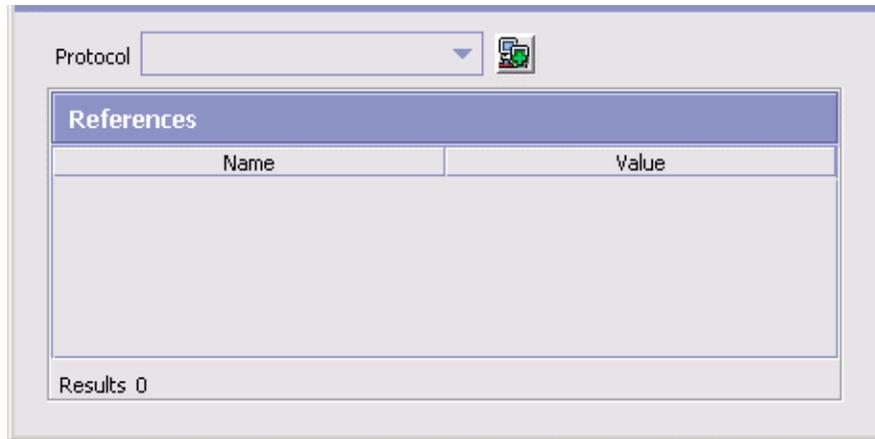


Table 3–32 Communication Protocol Details Pop-Up Window

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" . 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.

3.1.4.2 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 3–32](#) for field value descriptions.

4. Choose .

3.1.4.3 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 display.
2. From the Communication Protocols list, select the applicable communication protocol and choose .

3.1.4.4 Defining Buyer Documents

If you choose Buyer as the role for the organization, you may have to define the types of Buyer documents. Buyer documents are documents used by the Buyer organization when communicating with a Seller organization.

The following are some examples of Buyer documents:

- Planned Purchase Order
- Purchase Order Download
- Invoice

You can create, modify, and delete Buyer documents.

3.1.4.4.1 Creating a Buyer Document

To create a Buyer document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Buyer Documents tab. The Buyer Documents list displays.
3. Choose . The Buyer Document Detail pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–33](#) for field value descriptions.
5. Choose .

Table 3–33 Buyer Document Detail Pop-Up Window

Field	Description
Document	The business document to be used in your organization. For more information about business documents, see Section 10.3, "Defining Business Document Codes" .
Format	Select the format in which you want the document to appear. For example, XML or EDI. For more information about document formats, see Section 10.2, "Defining Document Format Codes" .
Protocol	Select the protocol you want to use when sending or receiving the document. For example, FTP.
Backup Protocol	Select a backup protocol to use in case there is a problem with the primary protocol. For example, set to e-mail service in case of occasional problems with the FTP site. For more information about protocols, see Section 3.1.4, "Defining Communication Protocols" .

Table 3–33 Buyer Document Detail Pop-Up Window

Field	Description
Integration Service	<p>Enter the service that the Buyer prefers to use to receive this document.</p> <p>For example, you are setting up an Invoice document. The Buyer prefers to receive this document from organizations using an EDI format. A particular organization that participates with the Buyer, also uses two additional Carriers that prefer to receive the document in formats other than EDI. The Enterprise has a router configured to look for the preferred service at the point in their pipeline where the Invoice document is sent to the Buyer. They also have several services configured to enable the transfer of documents in different formats, including EDI.</p> <p>You can use this field to specify the service to transfer the necessary information into EDI format at the necessary point in the pipeline.</p>
Reference 1	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 2	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 3	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 4	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 5	Enter any user-defined information. This field is disabled until you enter information and choose  .

3.1.4.4.2 Modifying a Buyer Document

To modify a Buyer document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Buyer Documents tab. The Buyer Documents list displays.
3. Select the applicable Buyer document and choose . The Buyer Document Detail pop-up window displays.

4. Modify information in the applicable fields. Refer to [Table 3–33](#) for field value descriptions.
5. Choose .

3.1.4.4.3 Deleting a Buyer Document

To delete a Buyer document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Buyer Documents tab. The Buyer Documents list displays.
3. Select the applicable Buyer document and choose .

3.1.4.5 Setting Up Seller Documents

If you choose Seller as the role for the organization, you may have to define the types of Seller documents. Seller documents are documents used by the Seller organization when communicating with a Buyer organization.

The following are some examples of Seller documents:

- Order Confirmation
- Shipment Confirmation
- Planned Order Modification

You can create, modify, and delete Seller documents.

3.1.4.5.1 Creating a Seller Document

To create a Seller document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Seller Documents tab. The Seller Documents list displays.
3. Choose . The Seller Document Detail pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–34](#) for field value descriptions.
5. Choose .

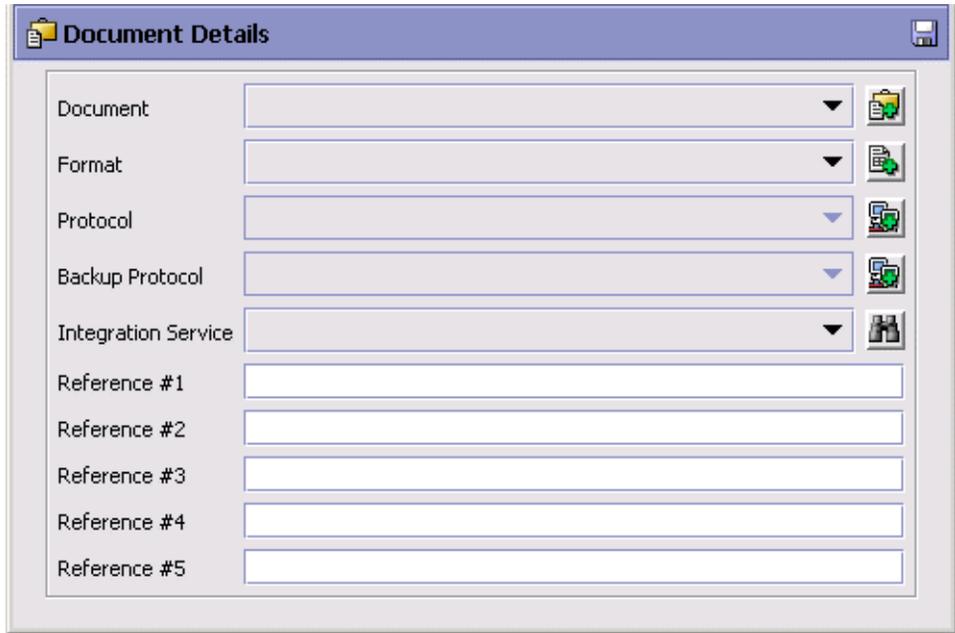


Table 3–34 Seller Document Detail Pop-Up Window

Field	Description
Document	The business document to be used in your organization. For more information about business documents, see Section 10.3, "Defining Business Document Codes" .
Format	Select the format in which you want the document to appear. For example, XML or EDI. For more information about document formats, see Section 10.2, "Defining Document Format Codes" .
Protocol	Select the protocol you want to use when sending or receiving the document. For example, FTP.
Backup Protocol	Select a backup protocol to use in case there is a problem with the primary protocol. For example, set to e-mail service in case of occasional problems with the FTP site. For more information about protocols, see Section 3.1.4, "Defining Communication Protocols" .

Table 3–34 Seller Document Detail Pop-Up Window

Field	Description
Integration Service	<p>Enter the service that the Seller prefers to use to receive this document.</p> <p>For example, you are setting up a Purchase Order document. The Seller prefers to receive this document from organizations using an EDI format. A particular organization that participates with the Seller, also uses two additional Sellers that prefer to receive the document in formats other than EDI. This organization has a router configured to look for the preferred service at the point in their pipeline where the Purchase Order document is sent to the Seller. They also have several services configured to enable the transfer of documents in different formats, including EDI.</p> <p>You can use this field to specify the service to transfer the necessary information into EDI format at the necessary point in the pipeline.</p>
Reference 1	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 2	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 3	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 4	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 5	Enter any user-defined information. This field is disabled until you enter information and choose  .

3.1.4.5.2 Modifying a Seller Document

To modify a Seller document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Seller Documents tab. The Seller Documents list displays.
3. Select the applicable Seller document and choose . The Seller Document Detail pop-up window displays.

4. Modify information in the applicable fields. Refer to [Table 3–34](#) for field value descriptions.
5. Choose .

3.1.4.5.3 Deleting a Seller Document

To delete a Seller document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Seller Documents tab. The Seller Documents list displays.
3. Select the applicable Seller document and choose .

3.1.4.6 Setting Up Carrier Documents

If you chose Carrier as the role for the organization, you may have to define the types of Carrier documents. Carrier documents are documents used in communicating carrier shipment information with Buyer and Seller organizations.

The following are some examples of Carrier documents:

- Package Details
- Load Tender Request
- Load Tender Confirmation

You can create, modify, and delete Carrier documents.

3.1.4.6.1 Creating a Carrier Document

To create a Carrier document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Carrier Documents tab. The Carrier Documents list displays.
3. Choose . The Carrier Document Detail pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–35](#) for field value descriptions.
5. Choose .

Table 3–35 Carrier Document Detail Pop-Up Window

Field	Description
Document	The business document to be used in your organization. For more information about business documents, see Section 10.3, "Defining Business Document Codes" .
Format	Select the format in which you want the document to appear. For example, XML or EDI. For more information about document formats, see Section 10.2, "Defining Document Format Codes" .
Protocol	Select the protocol you want to use when sending or receiving the document. For example, FTP.
Backup Protocol	Select a backup protocol to use in case there is a problem with the primary protocol. For example, set to e-mail service in case of occasional problems with the FTP site. For more information about protocols, see Section 3.1.4, "Defining Communication Protocols" .

Table 3–35 Carrier Document Detail Pop-Up Window

Field	Description
Integration Service	<p>Enter the service that the Carrier prefers to use to receive this document.</p> <p>For example, you are setting up a Package Type document. The Carrier prefers to receive this document from organizations using an EDI format. A particular organization that participates with the Carrier, also uses two additional Carriers that prefer to receive the document in formats other than EDI. The Enterprise has a router configured to look for the preferred service at the point in their pipeline where the Package Type document is sent to the Carrier. They also have several services configured to enable the transfer of documents in different formats, including EDI.</p> <p>You can use this field to specify the service to transfer the necessary information into EDI format at the necessary point in the pipeline.</p>
Reference 1	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 2	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 3	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 4	Enter any user-defined information. This field is disabled until you enter information and choose  .
Reference 5	Enter any user-defined information. This field is disabled until you enter information and choose  .

3.1.4.6.2 Modifying a Carrier Document

To modify a Carrier document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Carrier Documents tab. The Carrier Documents list displays.
3. Select the applicable Carrier document and choose . The Carrier Document Detail pop-up window displays.

4. Modify information in the applicable fields. Refer to [Table 3–35](#) for field value descriptions.
5. Choose .

3.1.4.6.3 Deleting a Carrier Document

To delete a Carrier document:

1. In the Organization Details window, choose the Communication tab. The Communication Protocols and Documents lists display.
2. From the Documents list, choose the Carrier Documents tab. The Carrier Documents list displays.
3. Select the applicable Carrier document and choose .

3.1.5 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 3–36](#) for field value descriptions.
3. Choose .

The screenshot displays the 'Payment Info' tab within a software application. The interface is organized into several sections:

- Primary Info:** Contains a text input field for 'Account Number With Hub'.
- Tax Information:** Includes a 'Tax Payer ID' text field, an 'Authority Type' text field, a 'Tax Exempt' checkbox, an 'Exemption Certificate' text field, and a 'Tax Jurisdiction' text field.
- Billing Address:** This section is expanded, showing a large text area containing the text 'AU'. Below this area are two sub-sections: 'Address' and 'Contact Info', each with a small icon.

Table 3–36 *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.

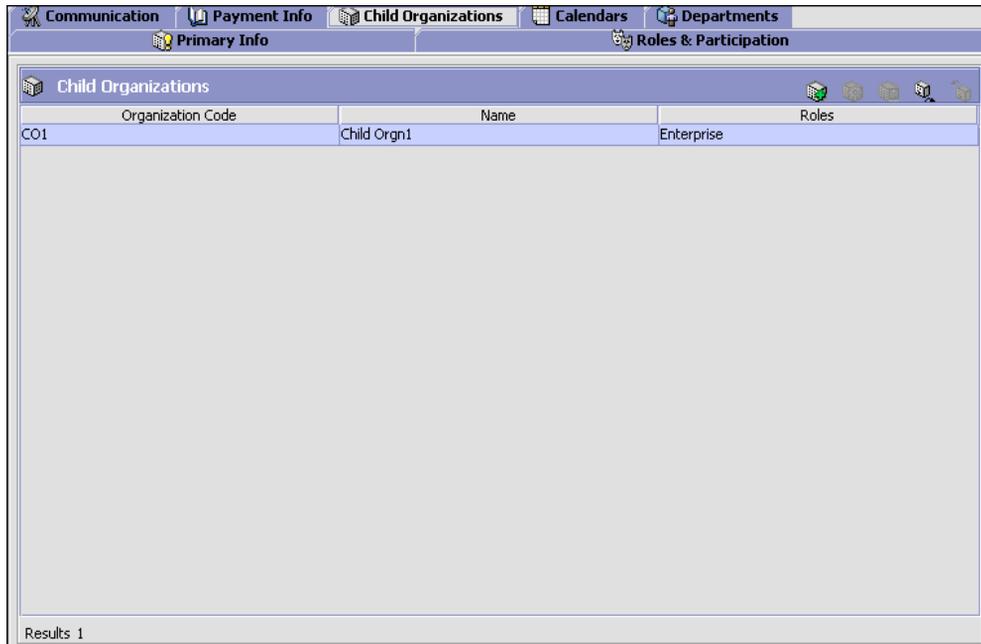
Table 3–36 Payment Info Tab

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	The the organization's billing address. This information is mandatory. Choose  to enter an address. Choose the Contact Info tab to view additional contact information.

3.1.6 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.



3.1.7 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.

Create Calendar

Inherit Definition From Calendar:

Calendar ID: Description:

Default Effective From: Default Effective To:

Shifts For Effective Period

Shift Name	Start Time	End Time	Sun	Mon	Tue	Wed	Thu	Fri	Sat
Morning	08:00	12:00		✓	✓	✓	✓	✓	
Afternoon	13:00	17:00		✓	✓	✓	✓		

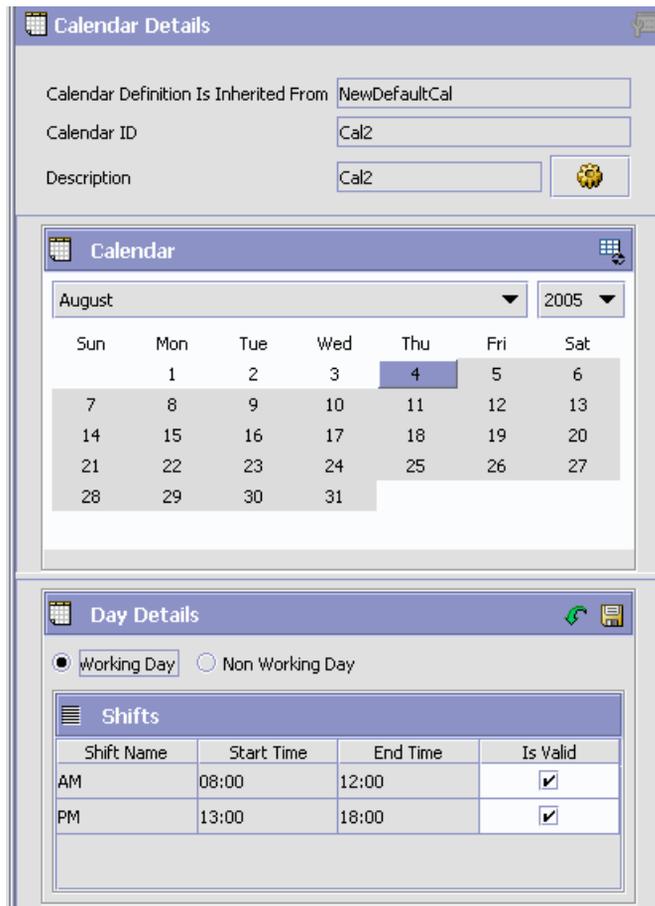
Table 3–37 Create Calendar Pop-up Window

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. However, because the ending date is perceived by the system as having the time 00:00:00, the ending date is not included in the date range.
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: NewDefaultCal

Calendar ID: Cal2

Description: Cal2

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](#)

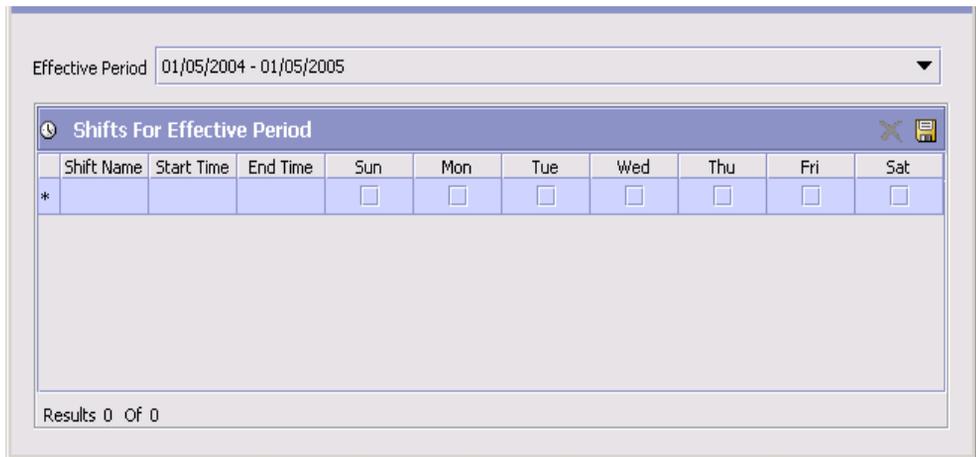
3.1.7.1 Defining a Calendar's Defaults

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 displays.



Shift Name	Start Time	End Time	Sun	Mon	Tue	Wed	Thu	Fri	Sat
*			<input type="checkbox"/>						

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 displays in Calendar Details pop-up window for any of the default days you have selected.

3.1.7.2 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.

The screenshot shows two windows from a configuration application. The top window, titled "Calendar Details", contains the following fields:

- Calendar Definition Is Inherited From: NewDefaultCal
- Calendar ID: Cal2
- Description: Cal2

The middle window, titled "Calendar", shows a calendar for August 2005. The date Thursday, August 4th, is selected.

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			

The bottom window, titled "Day Details", has two radio buttons: "Working Day" (selected) and "Non Working Day". Below it is a table titled "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.

3.1.8 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](#)

3.1.8.1 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 3–38](#) for field value descriptions.
4. Choose .

Table 3–38 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.
Queue	Select the default alert queue for the department to which the exceptions are routed. The list contains only those alert queues which are configured on the departments organization. For more information about creating alert queues, see, Chapter 12, "Configuring Alert Queues" .
Email Id	Enter the email id for the department.

3.1.8.2 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 3–38](#) for field value descriptions.

4. Choose .

3.1.8.3 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 .

3.1.9 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 displays 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.

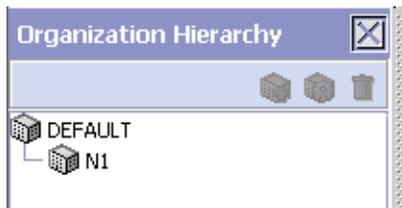
3.2 Creating and Modifying an Organizational Hierarchy

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 displays 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 displays 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 displays.
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 about defining inventory node type rules, see the *Sterling Global Inventory Visibility: 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 displays in the work area.
2. Choose . The Node Type Details pop-up window displays.
3. Enter information into the applicable fields. Refer to [Table 3–39](#) for field level descriptions.
4. Choose .



Node Type Details	
Node Type	Store
Description	Store

Table 3–39 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 displays in the work area.
2. Select the applicable node type and choose . The Node Type Details pop-up window displays.
3. Enter information into the applicable fields. Refer to [Table 3–39](#) 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 displays in the work area.
2. Select the applicable node type and choose .

4

Configuring Process Models

Process Modeling is the setting up of the Selling and Fulfillment Foundation business process workflow. The Selling and Fulfillment Foundation workflow consists of the entire set of business logic that defines how Selling and Fulfillment Foundation 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

Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation handles, and defines a common storage structure for all derived document types.

The following base document types are defined in Selling and Fulfillment Foundation:

- General
- Order
- Load
- 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 about document types, see [Table E-1, "Document Types"](#).

To complete a life cycle, 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 Selling and Fulfillment Foundation.

Following are the process types defined in Selling and Fulfillment Foundation (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
- Master Order Fulfillment
- 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 Selling and Fulfillment Foundation, 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 displays 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 displays.

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 displays in the document type tree with the associated process types.

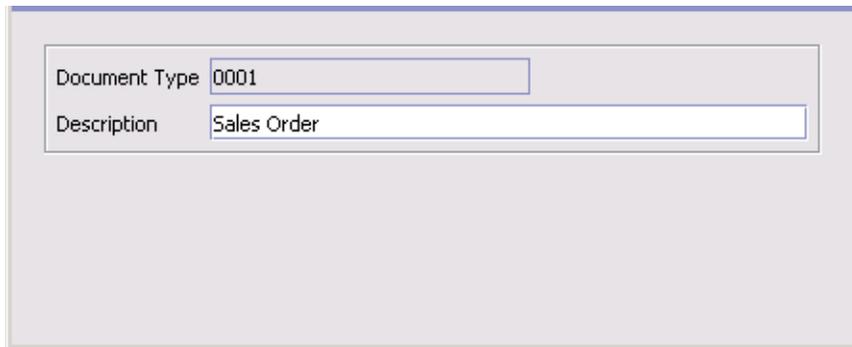
Note: An .ex extension is automatically appended to the document type and process type values you have specified.

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 displays.



The screenshot shows a window titled 'Document Type Details'. It contains two text input fields. The first field is labeled 'Document Type' and contains the value '0001'. The second field is labeled 'Description' and contains the value 'Sales Order'.

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 displays.

The screenshot shows the 'Primary Info' tab of the 'Process Type Details' window. The 'Process Type' is 'ORDER_FULFILLMENT' and the 'Process Type Name' is 'Order Fulfillment'. The 'Description' is 'Order Fulfillment' and the 'Document Classification' is 'Sales Order'. There are two checkboxes: 'Use As A Template Order' (unchecked) and 'Validate Item During Order Creation/Modification' (unchecked). The 'Driver Date' section has two radio buttons: 'Requested Ship Date' (unchecked) and 'Requested Delivery Date' (checked). Below this is the 'Order Creation' tab, which is currently selected. It contains three checkboxes: 'Use Template Order For Defaulting' (checked), 'Log Audits For Draft Order' (unchecked), and 'Default Delivery Method Based On Catalog' (checked). There are also two dropdown menus: 'Template Document Type' (set to 'Template Order') and 'Refund Fulfillment Order Document Type' (set to 'Sales Order'). A checkbox 'Suppress Audits For Change Order Status' is unchecked.

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 life cycle 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 displays.
3. Choose the Primary Info tab.

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

The screenshot shows a software configuration window with two main tabs: 'Primary Info' and 'Templates'. The 'Templates' sub-tab is selected. Under 'Primary Info', there are several input fields: 'Process Type' (text box with 'ORDER_FULFILLMENT'), 'Process Type Name' (text box with 'Order Fulfillment'), 'Description' (text box with 'Order Fulfillment'), and 'Document Classification' (dropdown menu with 'Sales Order'). There are also two checkboxes: 'Use As A Template Order' (unchecked) and 'Validate Item During Order Creation/Modification' (unchecked). To the right, under 'Driver Date', there are two radio buttons: 'Requested Ship Date' (unchecked) and 'Requested Delivery Date' (checked). Below this, the 'Order Creation' sub-tab is active, showing checkboxes for 'Use Template Order For Defaulting' (checked), 'Log Audits For Draft Order' (unchecked), and 'Default Delivery Method Based On Catalog' (checked). There are also dropdown menus for 'Template Document Type' (set to 'Template Order') and 'Refund Fulfillment Order Document Type' (set to 'Sales Order'), and a checkbox for 'Suppress Audits For Change Order Status' (unchecked).

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.
Document Classification	Select the document type you want to use for the process type.
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 units of measure to be validated against the Catalog Management application (or external Catalog Management application) upon order creation or adding an order line.

Table 4–1 Process Type's Primary Info Tab

Field	Description
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.
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>Sterling Distributed Order Management: User Guide</i>, the <i>Sterling Supply Collaboration: User Guide</i>, and the <i>Sterling 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>Catalog Management: Configuration Guide</i>.</p> <p>Note: This field should not be selected for any document types other than Sales Orders.</p>

Table 4–1 Process Type's Primary Info Tab

Field	Description
Suppress Audits for Change Order Status	Select this field if the CHANGE_STATUS audits have to be suppressed by the changeOrderStatus API for any change in the order status.
Inventory Tab	
Demand Type For Schedule	Select the demand type to be considered when the Schedule Order time-triggered transaction is run. 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>Sterling Global Inventory Visibility: Configuration Guide</i> . 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.
Demand Type for Release	Select the demand type to be considered when the Release Order time-triggered transaction is run. 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>Sterling Global Inventory Visibility: Configuration Guide</i> .
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. Note: This field is not applicable for procurement orders.

Table 4–1 Process Type's Primary Info Tab

Field	Description
Procurement Placed Supply Type	Select the supply type of the supply to be considered when a procurement order is placed.
Financials Tab	
Allow Invoice Creation	Select this field to enable invoice creation for orders or shipments.
Allow Pro Forma Invoice Creation For Shipments	Select this field to enable Pro Forma invoice creation for shipments. For more information about Pro Forma invoicing, see the <i>Selling and Fulfillment Foundation: Product Concepts Guide</i> . Note: This flag is only applicable to shipments created from orders. This flag is ignored for shipments without orders. For more information about creating Pro Forma invoices for shipments without orders, see the <i>Selling and Fulfillment Foundation: Javadocs</i> .
Allow Payment Processing	Select this field to enable payment processing. For more information about payment processing, see the <i>Selling and Fulfillment Foundation: Product Concepts Guide</i> .
Allow Price Calculation For Draft Orders	Select this field if you want to be able to associate a price with an item during draft order creation. For more information about configuring prices, see the <i>Sterling Distributed Order Management: Configuration Guide</i> or the <i>Business Center: Pricing Administration Guide</i> . 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 lists 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.

Table 4–1 Process Type's Primary Info Tab

Field	Description
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 lists 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	Select this field if you want the order document to have the ability to create chained orders during scheduling.
Chained Procurement Inbound Order Document Type	<p>Select the document type you want to use for the chained order document in a procurement inbound order scenario.</p> <p>The procurement inbound 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.</p>
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 Transfer Order Document Type	<p>Select the document type you want to use for the chained order document in a procurement transfer order scenario.</p> <p>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.</p>

Table 4–1 Process Type's Primary Info Tab

Field	Description
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 it was derived from.
Create Shipments For Products Being Delivered In Addition To Work Order	If this is checked, shipments should be created for all the products, which are delivered along with the work order. Note: If the flag is unchecked, you cannot create releases since releases are not required without shipments.

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

Document templates are used at various times throughout Selling and Fulfillment Foundation. 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 displays.
3. Choose the Templates tab. The available master templates for the fulfillment process type you are working with display as tabs. These master templates are retrieved from the `YFS_BASE_DOCUMENT_TYPE` table.

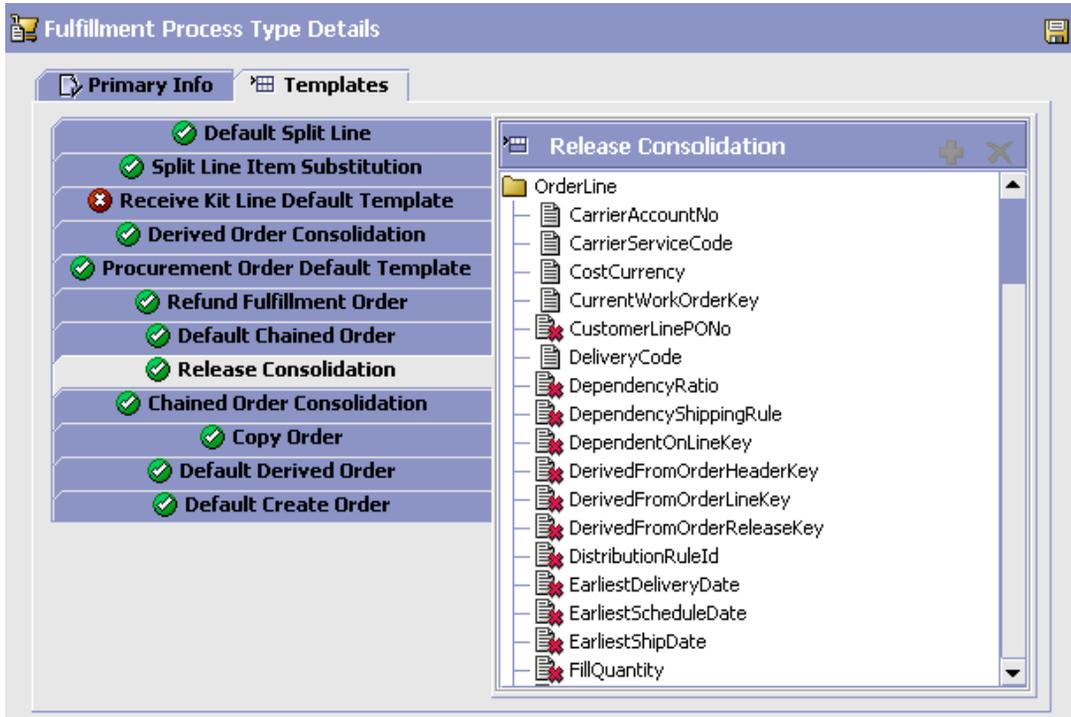
4. Choose the tab of the applicable master template. The master template XML is loaded from the YFS_BASE_DOCUMENT_TEMPLATE table and is combined with the template XML stored for this document type in the YFS_DOCUMENT_TEMPLATE table. Extended attributes are also added to the master template for each element 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

Selling and Fulfillment Foundation 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 display. 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 display 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 displays. If you are logged in as an Enterprise role, both the Hub Rule and My Rule components display. 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"](#).

4.2.2 Creating a Pipeline

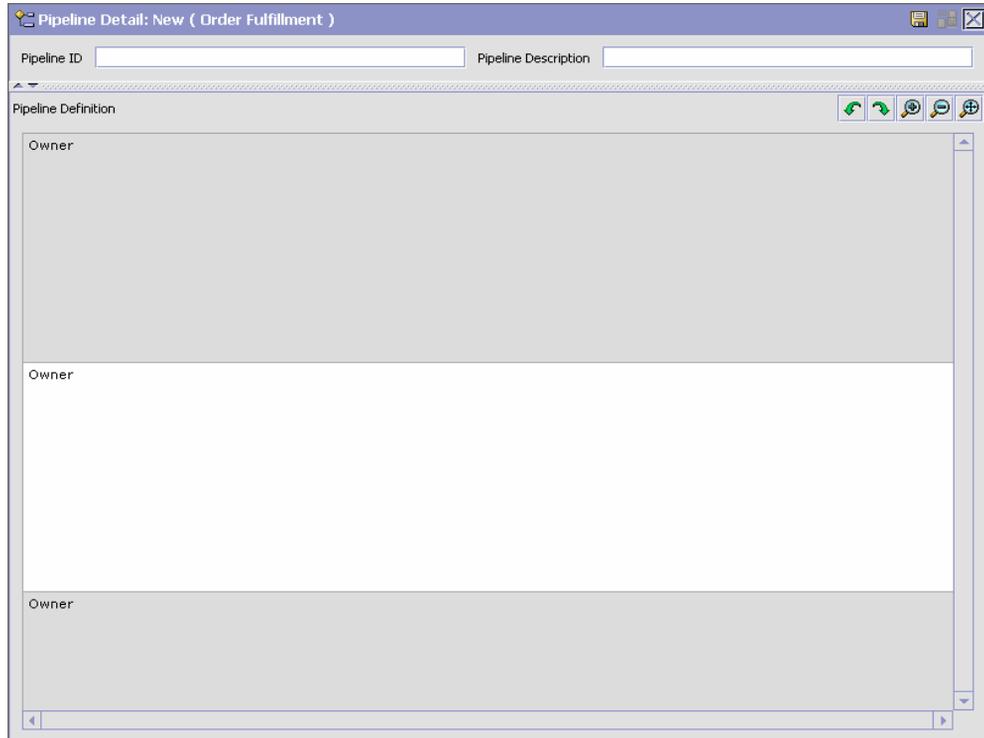
You can create a pipeline for the process type that you are working with.

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 display for the corresponding process type.
3. Choose the Pipeline tab.
4. Select Pipelines and choose . The Pipeline Detail window displays 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"](#).

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 display 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, Selling and Fulfillment Foundation creates 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.

Note: The Order Delivered Status can be picked up by more than one transaction. There is no implied order of processing between these transactions. Depending on which transaction is run first, this status is processed.

4.2.4 Modifying a Pipeline

Note: Selling and Fulfillment Foundation 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 the changes in your test, and then migrate the changes to your production environment using the Selling and Fulfillment Foundation 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 display 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 displays 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"](#).
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, Selling and Fulfillment Foundation creates 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

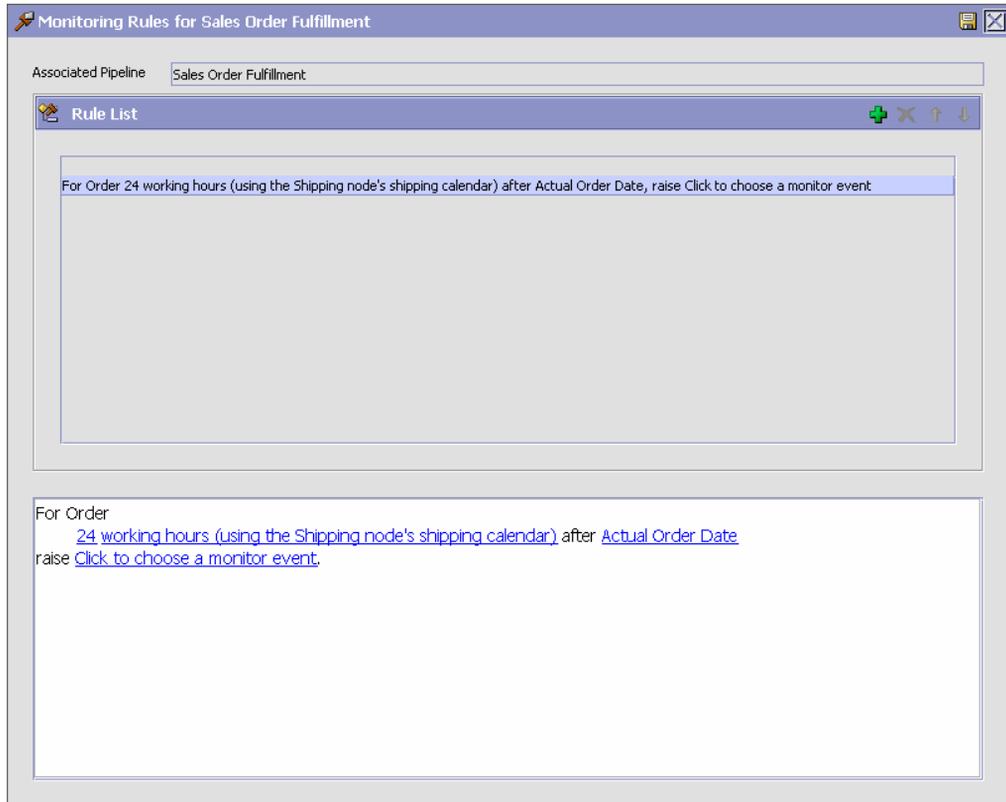
Using the monitoring rule components you configured while defining the process type, you can define the parameters used to monitor orders and shipments throughout their life cycle in fulfillment and shipment process type pipelines. For more information about configuring the monitoring rule components, see the *Sterling 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 display for the corresponding process type.
3. Choose the Pipeline tab.

- Expand the Pipelines branch.
- Select the applicable pipeline and choose . The Monitor Rules window displays in the work area.



- From Rule List, choose . The Rule Type pop-up window displays.



7. Select the rule type you want to define and add to the rule list. The rule details displays 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	If 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	If 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 .
Has been in Hold Type	If Order/Shipment has been in a specified Hold Type with a specified Hold Type Status for <i>n calendar hours</i> , then raise a monitor event .
Conditionally has been in Hold Type	If Order/Shipment meets a condition and has been in a specified Hold Type with a specified Hold Type Status for <i>n calendar hours</i> , then raise a monitor event .
Has been in Hold Type before a date	If Order/Shipment is in a specified Hold Type with a Hold Type Status with <i>n calendar hours</i> before a date , then raise a monitor event .
Conditionally has been in Hold Type before a date	If Order/Shipment meets a condition and is in a specified Hold Type with a Hold Type Status with <i>n calendar hours</i> before a date , then raise a monitor event .

Note: Hold-based monitoring rules can monitor holds that belong to the organization owning the pipeline.

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 Selling and Fulfillment Foundation. 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 Selling and Fulfillment Foundation, APIs are used to run transactions. When an API is invoked, the transaction ID is determined based on the context the API was run. The transaction ID identifies the transaction to be run. 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 *Selling and Fulfillment Foundation: Javadocs*.

Some extended transactions that are created may require custom coding to implement logic for the transaction. However, you can derive new transactions from the abstract transactions provided by Selling and Fulfillment Foundation. 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 the Service Definition Framework which calls a corresponding API within Selling and Fulfillment Foundation to run the transaction.

User-Triggered Transactions

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

Time-Triggered Transactions

A time-triggered transaction is invoked on scheduled intervals. In Selling and Fulfillment Foundation, 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 display for the corresponding process type.
3. Choose the Transactions tab.
4. Select the Transactions branch and choose . The Create New Transaction pop-up window displays.

Create New Transaction

When creating custom transaction, you can either create it as a derived transaction or a non-derived transaction. Sterling Supply Chain Applications provide 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 ▼

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

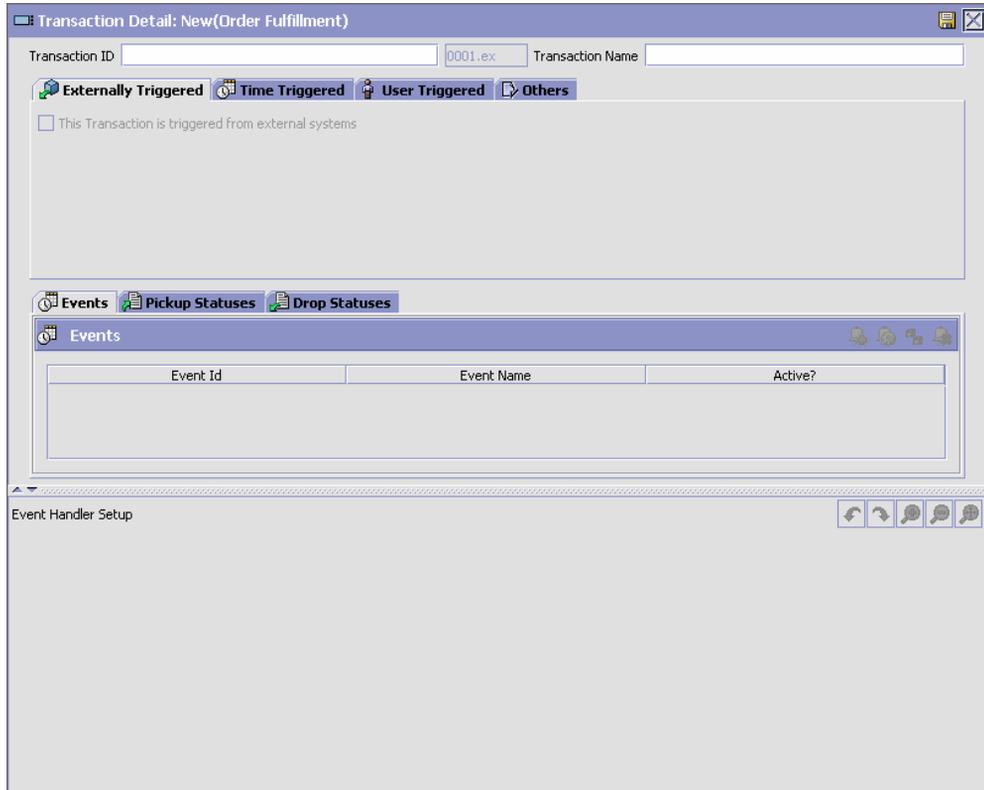


Table 4–3 Transaction Details Window

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" .

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" .
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" .
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). Note: This field is disabled for inherited transactions.
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 displays.

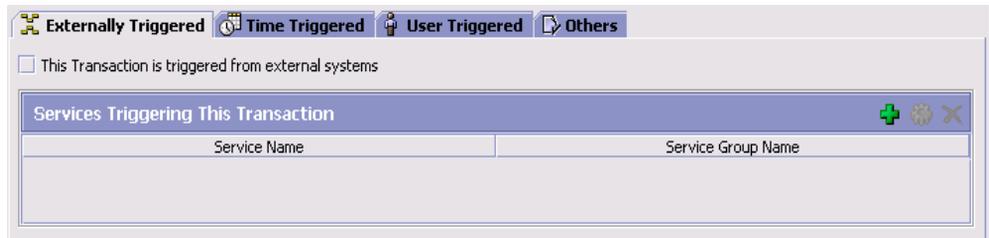
Table 4–3 Transaction Details Window

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" and Section 4.2.6.1.7, "Defining Event Handlers".</p>
Supports Transaction Dependency	<p>Check this box if you want to support dependency for this transaction. For more information about creating transaction dependency groups and any associated rules, see the <i>Sterling Distributed Order Management: Configuration Guide</i>.</p>
Supports Transaction Completion	<p>Check this box if you want to support transaction completion. For more information about configuring transaction completion for extended or custom transactions, see Section 4.2.6.1.12, "Configuring a Transaction Completion".</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".</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".</p>

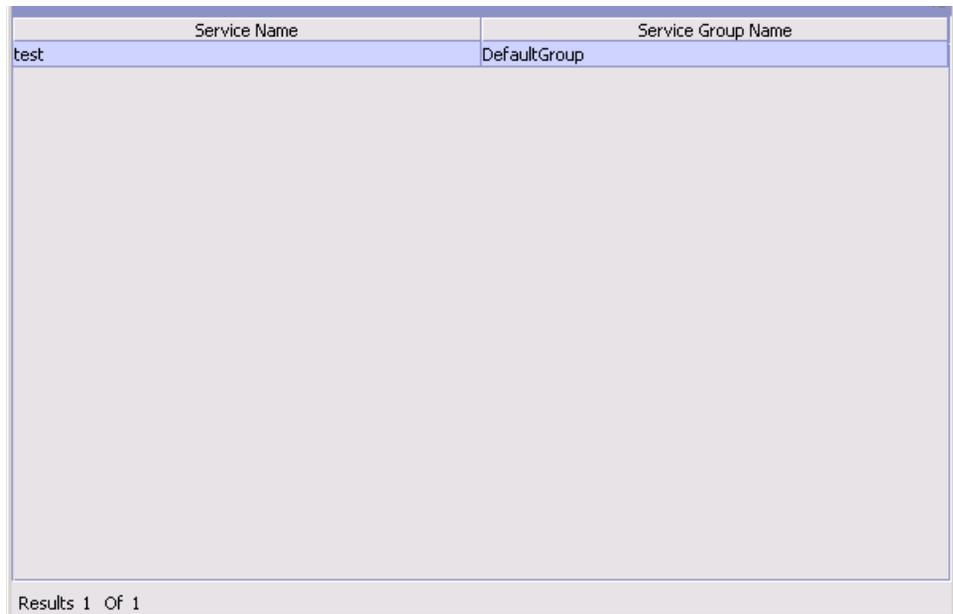
4.2.6.1.1 Specifying a Transaction as Externally-Triggered

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 in an externally-triggered transaction.
3. From the Services Triggering This Transaction list, choose . The Service List pop-up window displays.



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.
6. Choose .

4.2.6.1.2 Specifying a Transaction as Time-Triggered

You can set up a transaction to be triggered by the Selling and Fulfillment Foundation 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:

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



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

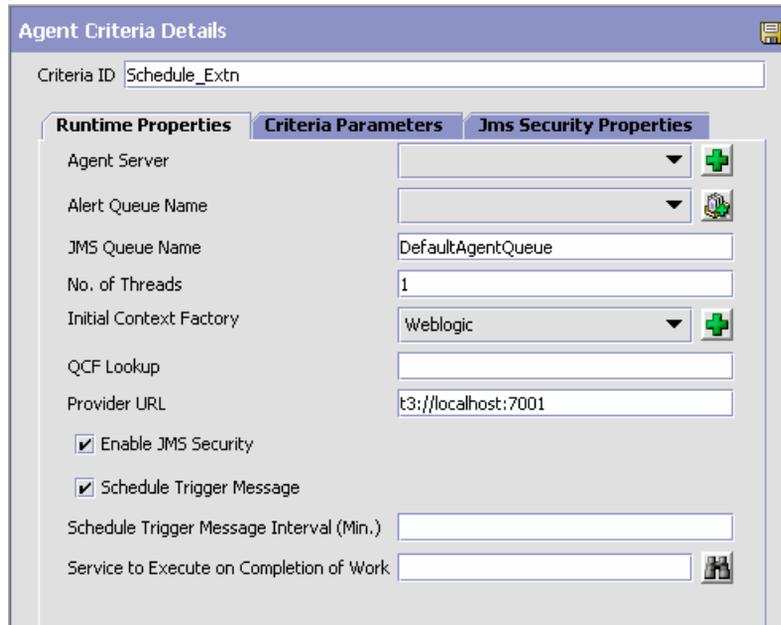


Table 4–4 Time-Triggered Transaction Runtime Properties

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>Selling and Fulfillment Foundation: 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>Select Jboss if you are using JBoss JMS. This sets the class name to <code>org.jnp.interfaces.NamingContextFactory</code>.</p> <p>If you defined an initial context factory code for ActiveMQ, select the entry you created for it. For more information about setting up other JMS vendors, such as ActiveMQ, which are not included in the default set, see Section A.2, "Configuring Communication Between an Agent and a JMS Server".</p> <p>Note: You can override this value by configuring the <code>yfs.agent.override.icf</code> property in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file. For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p> <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 Section 6.5, "Defining Initial Context Factory Codes".</p>

Table 4–4 Time-Triggered Transaction Runtime Properties

Control	Description
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 Selling and Fulfillment Foundation.</p> <p>Note: You can override this value by configuring the <code>yfs.agent.override.qcf</code> property in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file.</p> <p>For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
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> <p style="padding-left: 40px;"><code>t3://<IP address and port of the WLS instance></code></p> <p>If you use MQSeries through a JNDI file, enter the following value:</p> <p style="padding-left: 40px;"><code>file:/<pathname></code></p> <p>If you use MQSeries through WebSphere JNDI, enter the following value:</p> <p style="padding-left: 40px;"><code>corbaloc::<DNS server name or IP address>:<bootstrapport></code></p> <p>If you use JBoss JMS, enter the following value:</p> <p style="padding-left: 40px;"><code>jnp://<IP address and port of the JBoss instance></code></p> <p>Note: You can override this value by configuring the <code>yfs.agent.override.providerurl</code> property in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file.</p> <p>The Provider URL format is:</p> <p style="padding-left: 40px;"><code>t3://<ip Address>:<port></code></p> <p>For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>

Table 4–4 Time-Triggered Transaction Runtime Properties

Control	Description
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 configure Queue and/or JNDI based JMS security.</p> <p>Note: You can override this value by configuring the <code>yfs.agent.override.auth.enabled</code> property in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file.</p> <p>For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
Schedule Trigger Message	<p>Check this box to configure the agent to run 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)	Enter the desired time interval in minutes.
Service to Execute on Completion of Work	Enter the service to be run upon completing the execution of the selected agent. You can select the required service by using the  button.
Criteria Parameters	
Parameter Name	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" .
Parameter Value	The value of the parameter sent to the transaction. For valid names and parameters, see Appendix A, "Time-Triggered Transaction Reference" .

Table 4–4 Time-Triggered Transaction Runtime Properties

Control	Description
<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>Note: You can override the JMS security parameters (userid and password) by configuring the following parameters in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file:</p> <pre>yfs.agent.override.auth.userid yfs.agent.override.auth.password</pre> <p>For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p> <p>For more information about the application server-specific security mechanisms, see Setting up the JMS Security Properties.</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 Queue Based Security and/or JNDI Based Security

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

- For Queue Based Security set the following parameters:

- `sci.queuebasedsecurity.userid=<username configured in the APPLICATION_SERVER and assigned to the queue>`
- `sci.queuebasedsecurity.password=<password for the above username as configured for the APPLICATION_SERVER>`

Note: JBoss does not support queue based security for JMS service. Only JNDI based security is supported.

- For JNDI Based security set the following parameters:
 - `java.naming.security.principal=<user ID configured in the APPLICATION_SERVER and assigned to the JNDI >`
 - `java.naming.security.credentials=<password for the above user ID as configured for the APPLICATION_SERVER>`

Note: For more information about the authentication mechanism, setting up queues and QCF, refer to individual Application Server's 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 modifying, ensure that 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\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 about 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: Refer to the WebSphere MQ documentation to learn how to turn on the SSL on the server channel to which the Selling and Fulfillment Foundation agents and services connect. For more information about how to use the SSLCIPHERSPEC option while creating the QCF, see the IBM documentation.

For JBoss, before modifying, ensure that you have added following jars to the CLASSPATH:

- JBOSS_HOME refer to the installation directory of the JBoss software
- 1. To allow agents to be authenticated to JBoss JNDI, add the following definitions:
 - -Djava.ext.dirs=<CLASSPATH>, where the CLASSPATH should contain the following directories:
 - * <JBOSS_HOME>/client/jbossall-client.jar
 - * <JBOSS_HOME>/server/<server-home>/jboss-aop-jdk50.deployer/jboss-aop-jdk50.jar
 - * <JBOSS_HOME>/jboss-messaging-client.jar

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 the 'User Triggered' tab selected in the Transaction Details window. The tab bar at the top includes 'Externally Triggered', 'Time Triggered', 'User Triggered', and 'Others'. Below the tabs, there is a checkbox labeled 'This transaction is triggered by Users' which is currently unchecked. Below the checkbox is a text field labeled 'When ready, notify user using' with a dropdown arrow on the right side.

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 displays.
3. Enter information in the applicable fields. Refer to [Table 4-6](#) for field level descriptions.
4. Choose OK.

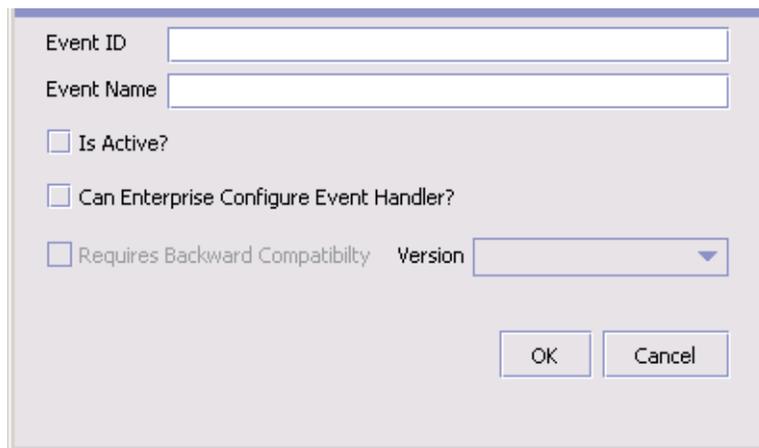


Table 4–6 Event Details Pop-Up Window

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. Note: If the transaction works across multiple enterprises or enterprise information is not available to the transaction, the default event handler is used.
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 displays.
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

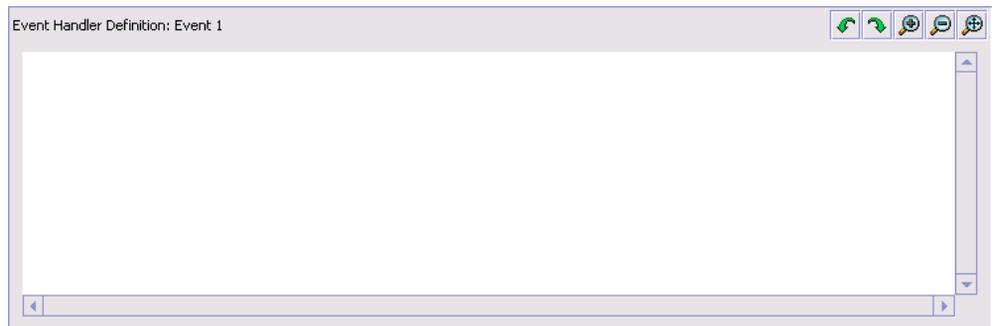
You can define event handlers that determine the types of actions that are performed when an event in a transaction occurs. You can provide conditions that apply to the event handler.

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 *Selling and Fulfillment Foundation: 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 .

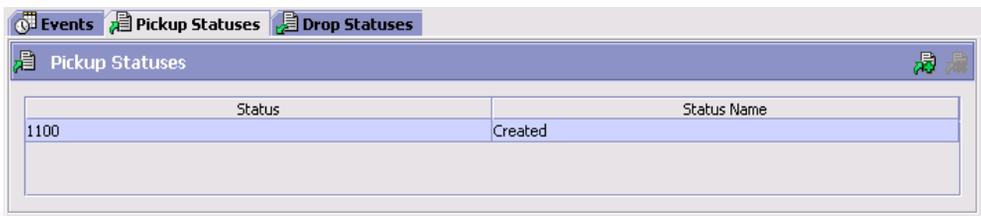
4.2.6.1.8 Adding a Pickup Status to an Extended Transaction

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 displays.



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.

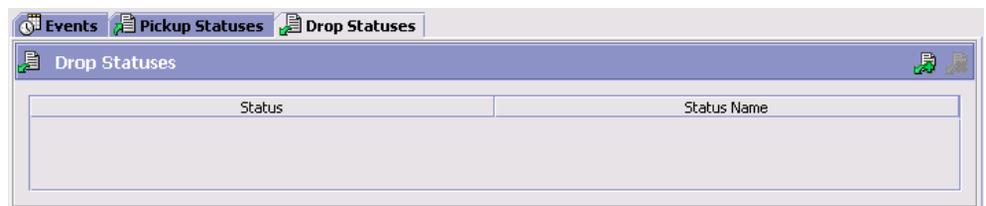
4.2.6.1.10 Adding a Drop Status to an Extended Transaction

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 displays.



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 Configuring a Transaction Completion

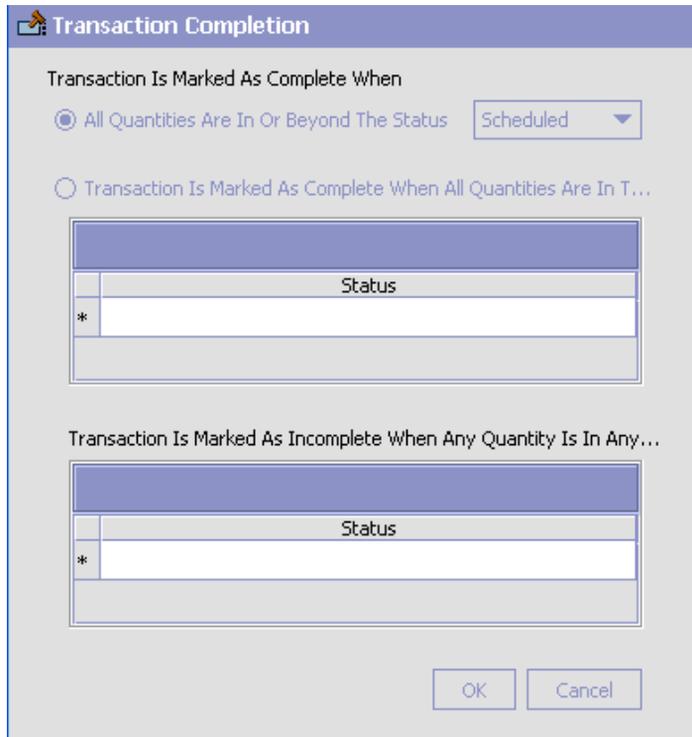
You can configure the transactions to be completed based on an order line status in the order fulfillment process type. The completion criteria can be defined only for custom or extended transactions. When an order runs these transactions, it is evaluated for completion. Upon evaluation the transaction it is marked as complete or incomplete as defined in the transaction completion criteria. This configuration provides you the flexibility to set up completion dependencies for transaction-based on its drop statuses.

Note: You cannot configure transaction completion for the standard transactions provided by Selling and Fulfillment Foundation. For example, you cannot configure completion for the standard `Schedule` transaction.

However, some of the status-based transactions such as `Schedule` and `Release` are provided with completion criteria and can be viewed by selecting  in the drop status tab.

Apart from transactions, completion criteria can also be configured for the extended listeners used in the pipeline. However, you need to configure the completion for every instance of the listener.

For more information about the concepts of transaction dependencies, see the *Selling and Fulfillment Foundation: Product Concepts Guide*. For information about configuring transaction dependencies for the order fulfillment model, see the *Sterling Distributed Order Management: Configuration Guide*.



The image shows a dialog box titled "Transaction Completion" with a small icon of a document with a checkmark. It contains two radio button options for when a transaction is marked as complete. The first option, "All Quantities Are In Or Beyond The Status", is selected and has a dropdown menu set to "Scheduled". The second option, "Transaction Is Marked As Complete When All Quantities Are In T...", is unselected. Below each option is a table with a header "Status" and a single row with an asterisk "*" in the first column and an empty text field in the second column. At the bottom right are "OK" and "Cancel" buttons.

Transaction Completion

Transaction Is Marked As Complete When

All Quantities Are In Or Beyond The Status Scheduled ▼

Transaction Is Marked As Complete When All Quantities Are In T...

Status	
*	

Transaction Is Marked As Incomplete When Any Quantity Is In Any...

Status	
*	

OK Cancel

To configure the transaction completion setup:

1. From the Transaction Detail window, choose the Drop Statuses tab.
2. Choose  to configure the transaction completion. The Transaction Completion window displays.
3. Enter information into the applicable fields. Refer to [Table 4–7, "Transaction Completion Configuration Details"](#) for field value descriptions.
4. After entering the details choose OK.

Table 4–7 Transaction Completion Configuration Details

Field	Description
Configure Transaction Completion For Order Line	Check this box if you want to configure the transaction completion based on an order line status.
Transaction is Marked As Complete When	
All Quantity Are In or Beyond the Status	Select this option if you want to configure the transaction when the order line is beyond the status specified in the drop-down. The transaction is considered incomplete if any quantity is below the status that is configured.
Transaction is Marked as Complete When All Quantities Are In the Following Statuses	Select the transaction completion status from the drop-down option. The drop down list all statuses defined for that organizations. Use the  icon to select from a list of available drop statuses.
Transaction is Marked as Incomplete When Any Quantity Is in Any of the Following Statuses	Select the status from the drop-down for which the transaction is considered incomplete. Use the  icon to select the status from the list of available statuses.

For deleting the chosen statuses for the completion or incompleteness of the transaction, choose the appropriate status and click .

TRANSACTION_DEPENDENCY.READY_TO_PROCESS event is triggered when the transaction is completed to notify the user that the order is ready to process the dependent transaction. This event also enables custom transactions to know that the order is ready for processing.

Note: The READY_TO_PROCESS event is triggered only when a transaction is ready for immediate processing.

You can have multiple transactions that could become ready due to the completion of one transaction. In this case, each transaction is output along with the order lines that are available.

The lines that become available due to dependency removal are published in the event.

4.2.6.1.13 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 the Selling and Fulfillment Foundation transactions can invoke them to perform the necessary logic at runtime.

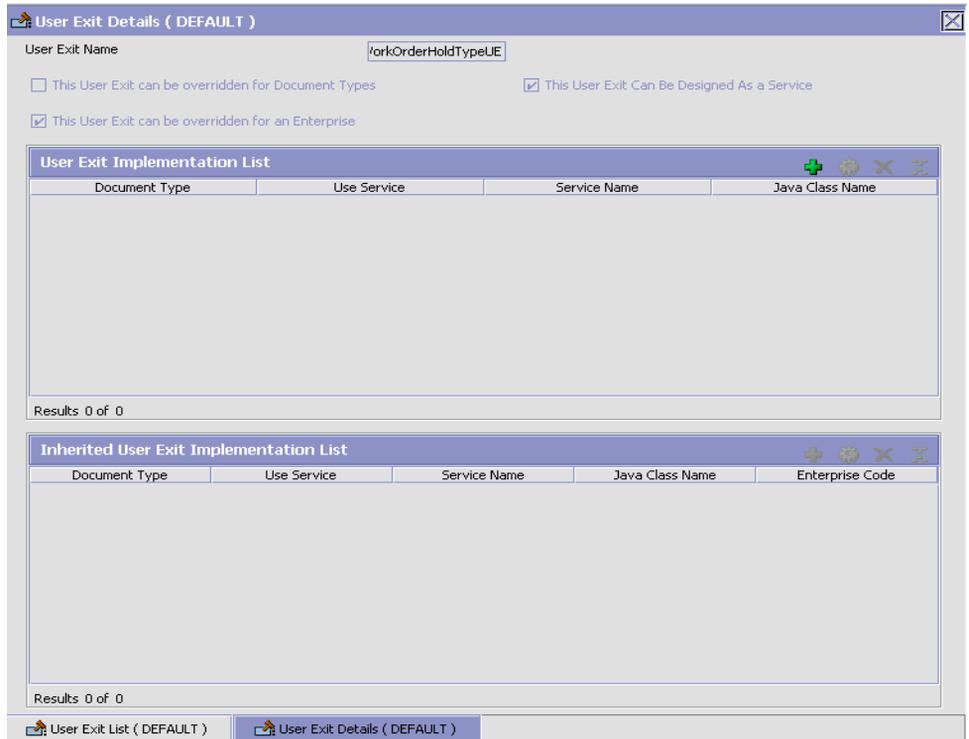
To manage a base transaction's user exits:

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 displays.



3. From the User Exit Implementation List table, choose . User Exit Implementation Details displays.
4. Enter information into the applicable fields. Refer to [Table 4–8](#) 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 4–8 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 4–8 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 displays 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 Selling and Fulfillment Foundation 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 are waiting 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 cause an error.
Wait Time (seconds)	Time for which the user exit call waits in queue. If the wait time exceeds the configured wait time, an exception is 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.

Note: When creating a transaction that is derived from an abstract transaction, completing the document field type is not mandatory, but is recommended.

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 display for the corresponding process type.
3. Choose the Transactions tab.
4. Select the Transactions branch and choose . The Create New Transaction pop-up window displays.
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 displays 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 transaction details screen for a listener allows multiple drop statuses to be added. A validation is performed to prevent the removal of a

transaction drop status record if there is a pipeline listener record for that drop status. When setting the statuses listened to in the pipeline, specify which drop status to use for each listened to status. When a child order changes status, the parent order is updated with the specific drop status for the status to which the child order has just changed.

This listener listens to all the same statuses as the previous three listeners combined and drops into whichever drop status that the previous three listeners did.

Listeners that change status on an order raise an event upon status change. Listeners raise the event for each order line that has a status change. The ON_STATUS_CHANGE event is raised for each order line using the Listener transaction.

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 display for the corresponding process type.
3. Choose the Transactions tab.
4. Select the Transactions branch and choose . The Create New Transaction pop-up window displays.
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 displays.

Listener Transaction: Status Change Listener

Process Type Listening To: [Empty Dropdown]

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 displays.
12. Select the statuses you want the listener to track and choose .
13. Choose OK.

Note: Work Order Status Listeners do not work on provided service or delivery service work orders.

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 display 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 displays 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 display 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.

Selling and Fulfillment Foundation 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.

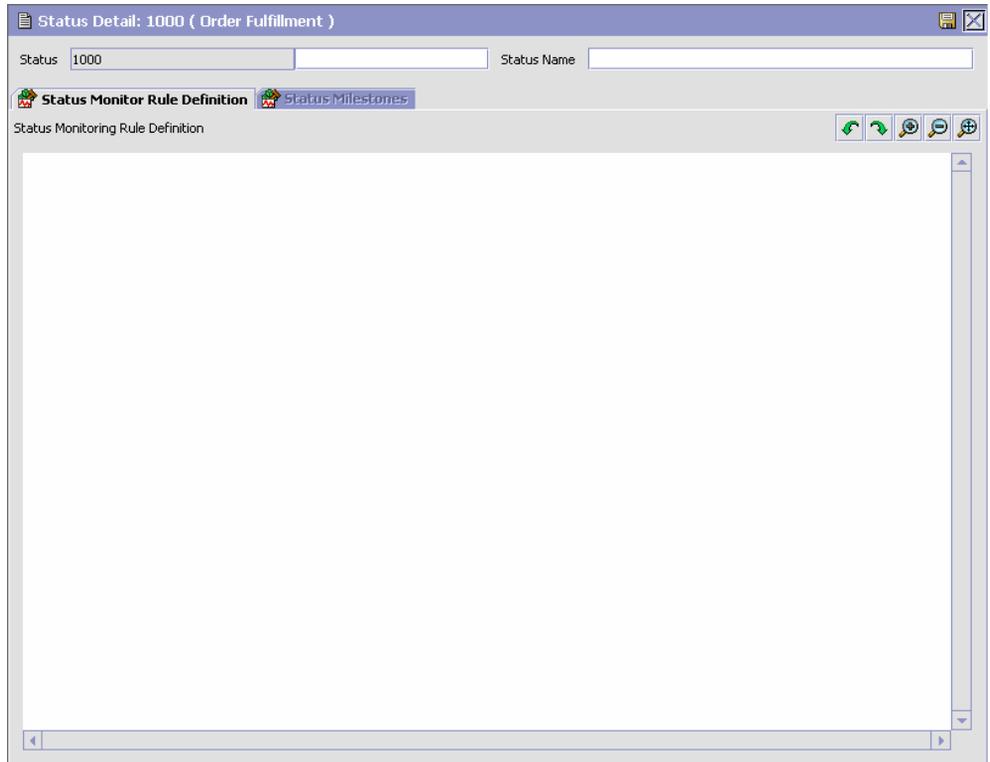
You can use the Statuses tab for:

- [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 display 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 displays 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 display 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 displays 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 display 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.

Note: The following setup for the status monitoring rule definition is for the order monitor.

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 display 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 displays in the work area.
6. Right-click in the work area and select Actions > Add Monitor Node. A monitor node displays 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 displays 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 is therefore disabled:

- Count Execution
 - General
 - Load Execution
 - Manifesting
 - Move Request Execution
 - Outbound Picking
 - Outbound Shipment
 - 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 Selling and Fulfillment Foundation. You can use these attributes in any combination or you can create conditions that run 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.

Static Conditions

The behavior of a static condition differs between different flows (for example, SDF, Pipeline, Pipeline Determination Rule, and Event Handlers) based on the data that is available for condition evaluation.

- In SDF, the entire flow or input data is available for condition evaluation.
- In case of a Pipeline Determination Rule, only a limited set of published data (per process type) is available for condition evaluation. For the complete list of out-of-the-box process types, refer to *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
- In Pipeline or Events, only a limited set of published data (per entity or event) is available for condition evaluation. For more information on Keydata of the concerned event, refer to *Selling and Fulfillment Foundation: Javadocs*.
- The Enter your own attribute configuration is applicable only for unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter. Also, this configuration has limited visibility to the contextual condition and hence it is not re-usable. When there is a need for re-usability across conditions, `extn_conditionbuilder.xml` should be used which forces the metadata for condition builder configurations.

Dynamic Conditions

Dynamic conditions provide complete visibility into the incoming flow data and flexibility for evaluating any kind of simple or complex conditions.

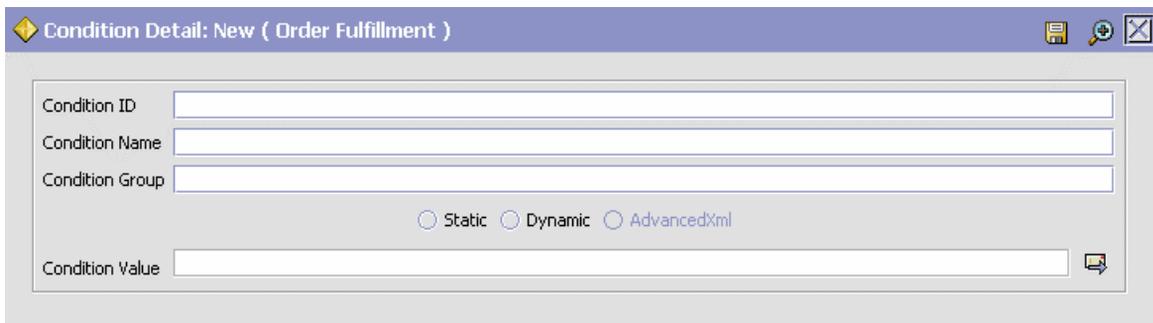
You can use the Conditions tab for:

- [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 display for the corresponding process type.
3. Choose the Conditions Tab.
4. Expand the Conditions branch.
5. Choose . The Condition Details window displays in the work area.
6. Enter information in the applicable fields. Refer to [Table 4–9](#) for field level descriptions.
7. Choose .



Condition Detail: New (Order Fulfillment)

Condition ID

Condition Name

Condition Group

Static Dynamic AdvancedXml

Condition Value

Table 4–9 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.
Static	If this is checked you must enter a condition value for the static condition.
Dynamic	If this is checked you must enter a Java class name that evaluates the condition at runtime.
AdvancedXML	<p>If you are creating a new condition, this option is disabled as a new condition of the advanced XML type must be created using the Sterling Greex Editor IDE tool. For more information about creating an advanced XML condition using the Sterling Greex Editor, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>This option is automatically selected whenever you modify a condition of the advanced XML type.</p>
Condition Value (if Static is checked)	<p>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.</p> <p>You can enter your own attribute or an extended attribute if Static condition is checked. For more information about creating these attributes, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p>

Table 4–9 Condition Details Window

Field	Description
Class Name (if Dynamic is checked)	<p>Enter the class name that implements the following Java interface:</p> <pre>com.yantra.ycp.japi.YCPDynamicCondition</pre> <p>Note: To use extended attributes for a condition, implement the YCPDynamicConditionEx interface. For more information about implementing this interface, refer to the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i></p>
Condition Properties (if Dynamic is checked)	<p>Specify the custom name or value properties which are set into the condition evaluating java class file before evaluating the condition. For more information about creating custom attributes, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p>

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 about implementing custom attributes and incorporating them in the condition builder, see the *Selling and Fulfillment Foundation: Extending the Condition Builder 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 display 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 displays in the work area.
7. Enter information in the applicable fields. Refer to [Table 4–9](#) the [Transaction Details Window](#) table for field level descriptions. (Refer to [Section 4.2.8.1.1, "Using the Condition Builder"](#) for information about modifying a static condition). (Refer to [Table 4–10](#) for field level descriptions for an advanced XML condition).
8. Choose .

Table 4–10 Advanced XML Condition Details Window

Field	Description
Condition ID	Enter the new identifier for the advanced XML condition (if required).
Condition Name	Enter the new name for the advanced XML condition (if required).
Condition Group	Enter the new name for the advanced XML condition's group (if required). Condition Group allows you to group related conditions within the condition tree.
Advanced XML (if AdvancedXML is checked)	<p>This screen displays only when you are editing an advanced XML condition that is created using the Sterling Greex Editor IDE tool. For more information about creating an advanced XML condition using the Sterling Greex Editor, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i>. This screen describes an advanced XML condition in simple English. All modifiable parameters of an advanced XML condition display as a hyperlink on the screen.</p> <p>Click on the hyperlink of the parameter whose value you want to edit. Specify the new value for the parameter in the pop-up screen. The pop-up screen displays the old value. You can also enter the new value. The new value reflects in the Advanced XML screen as well as in the Screen View screen when you click on the save button in the pop-up screen.</p>

Table 4–10 Advanced XML Condition Details Window

Field	Description
Source View (if AdvancedXML is checked)	This screen displays only when you are editing an advanced XML condition that is created using the Sterling Greex Editor IDE tool. This screen displays a specific advanced XML condition in an XML form or as defined in the advanced XML file. You cannot make changes to any parameter of the advanced XML condition.
Condition Cases (if AdvancedXML is checked)	Specify the custom cases for the decision table-based advanced XML condition. For more information about creating custom cases for a decision table-based advanced XML condition, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .

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 display 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 display 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 displays in the work area.
7. Choose . The Entities Affected by this Condition pop-up window displays. 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 display for the corresponding process type.
3. Choose the Actions Tab.
4. Expand the Actions branch.
5. Choose . The Action Details window displays.
6. Enter information in the applicable fields. Refer to [Table 4–11](#) 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 4–11 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 appear together in the Action tab in the Process Modeling tree.
Invoked Services	

Table 4–11 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, which 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, WebSphere, or JBoss):</p> <ul style="list-style-type: none"> • \$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
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 4–11 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	Select this field if you want to call a particular Java component.
Class Name	Enter the Java class name.
Execute	Select this field if you want to call a particular executable.
Program	Enter the Program (executable) name. Make sure the executable exists in the system PATH.

Table 4–11 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 are available in the drop-down. Any actions created from this screen using the create button are 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 display 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 displays.
7. Enter information in the applicable fields. Refer to [Table 4–11](#) 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 display 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 Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation 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. Selling and Fulfillment Foundation 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. Selling and Fulfillment Foundation supports the following asynchronous transport types:

- Asynchronous MQ JMS Queue
- Asynchronous WebLogic JMS Queue
- Database
- File IO
- FTP
- Generic JMS
- 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. Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation Adapter with an external system.

Selling and Fulfillment Foundation 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 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 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 display 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 displays.
5. Enter information in the applicable fields. Refer to [Table 4–12](#) for field value descriptions.
6. Choose OK. The Service Details Window displays.

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.

Create New Service

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 Sterling Selling and Fulfillment Suite

In an Asynchronous Mode
 In a Synchronous Mode

Table 4–12 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 Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation.	

Table 4–12 Create New Service Window

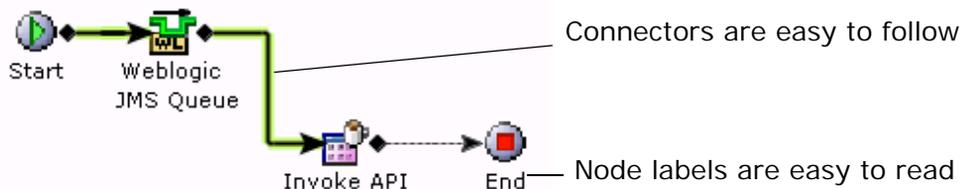
Property	Description
In an Asynchronous Mode	<p>Select this option when the service must start by retrieving a message from an asynchronous transport source.</p> <p>The service starts from a queue or database. The definition of the service does not need detail how the message arrives at asynchronous source such as queue or database.</p> <p>When this option is selected the first node after the start node must be an asynchronous transport node.</p>
In a Synchronous Mode	<p>Select this option when the service is invoked from Selling and Fulfillment Foundation or through an API synchronously.</p>

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 you 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 that 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 display 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 displays.
6. In the work area, choose the applicable node, its properties panel displays 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 display 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 display 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 displays.
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.

The Applications Manager's **Security Management** is used to create users, user groups, and teams. Once these have been created, permissions can be assigned.

You can use the Security branch for:

- [Defining Users](#)
- [Defining User Groups](#)
- [Defining Teams](#)
- [Defining Data Access Policies](#)
- [Configuring API Security](#)

5.1 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.

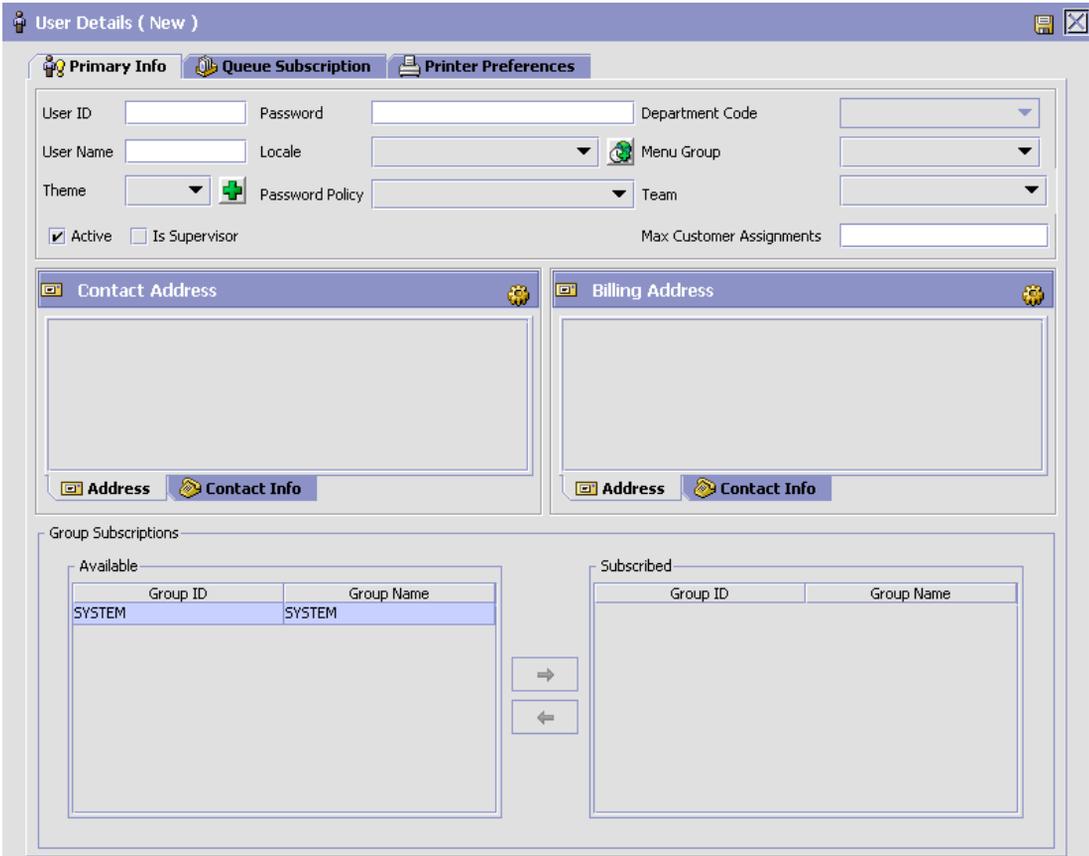
You can use the Users branch for:

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

5.1.1 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 displays in the work area.
2. Choose . The User 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.



User Details (New)

Primary Info Queue Subscription Printer Preferences

User ID Password Department Code

User Name Locale Menu Group

Theme Password Policy Team

Active Is Supervisor Max Customer Assignments

Contact Address **Billing Address**

Address **Contact Info** **Address** **Contact Info**

Group Subscriptions

Available

Group ID	Group Name
SYSTEM	SYSTEM

Subscribed

Group ID	Group Name
----------	------------

→
←

Table 5–1 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.
Department Code	Select the code of the department to associate with the user.
User Name	Enter the user's name.
Locale	Select the locale the user is located in. 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.
Menu Group	Select the menu group representing the menu options you want users to see when they log into the Application Console.
Theme	Enter the available theme as you want it to appear for the user. The theme determines how the color scheme of the Console and Applications Manager displays to the user. The available themes are: <ul style="list-style-type: none"> • Earth • Jade • Sapphire Note: You can extend the system to include as many themes as you want.
Password Policy	Select the password policy that you want to associate with the user. For additional information about password policies, see the <i>Selling and Fulfillment Foundation: Password Policy Management Guide</i> .

Table 5–1 User Details Window

Field	Description
Team	<p>Select the team to which you want to assign the user. For more information about teams, see Section 5.3, "Defining Teams".</p> <p>Notes:</p> <ul style="list-style-type: none"> The list contains all the teams configured for the organization to which the user belongs and the parent organization, if applicable. Clearing of the database cache is required to take into effect the changes made to this configuration.
Active	Check this if the user is currently active in the organization. Inactive users cannot log in.
Is Supervisor	Check this if the user is a personnel supervisor.
Max Customer Assignments	Enter the maximum number of customers that this user can be assigned to manage.
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	
Available	<p>A list of the available user groups.</p> <p>To subscribe a user to a user group, select the applicable user group and choose .</p>
Subscribed	<p>A list of the user groups to which the user is subscribed.</p> <p>To remove a user from a user group, select the applicable user group and choose .</p>

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 > Application Platform. The Application Platform tree displays in the side panel.
2. From the Application Platform tree, choose Security > Users. The User Search window displays 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 5–1](#) for field value descriptions.
6. Choose .

5.1.2 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.

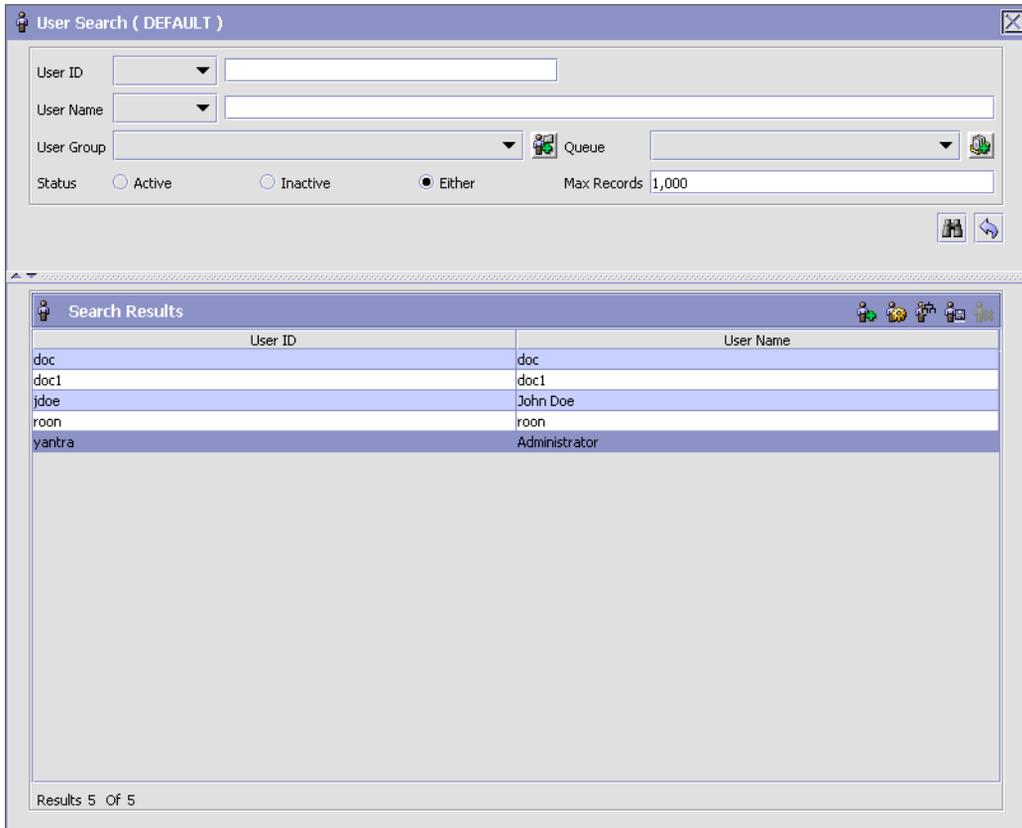
The association of a printer to a station overrides the group preference of the specified user. The station is a static location where devices may be directly attached to a station.

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 > Application Platform. The Application Platform tree displays in the side panel.

2. From the tree in the application rules side panel, choose Security > Users. The User Search window displays.
3. In the User Search window, enter applicable search criteria.
4. Choose . The list of users displays in the Search Results panel of the User Search window.



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 displays.
7. In the User Details window, choose the Printer Preferences tab. The Printer Preferences tab window displays.

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

For more information about Setting Up a User (Creating, Modifying, or Deleting a User), see [Section 5.1, "Defining Users"](#).

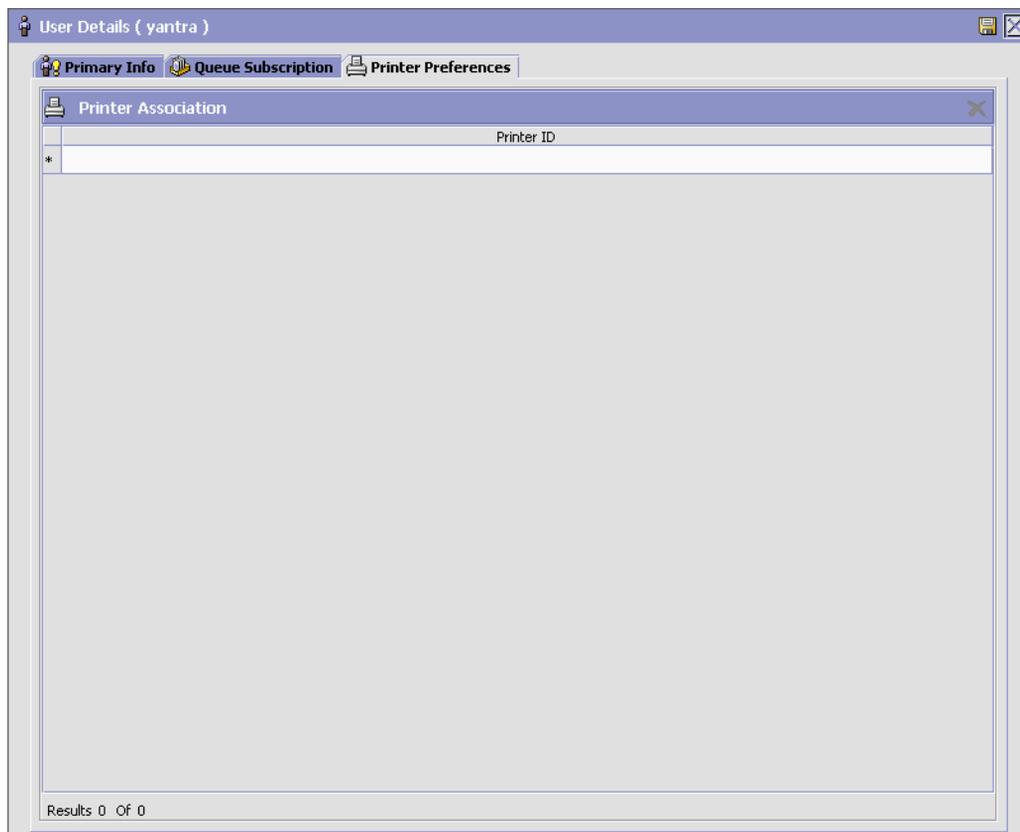


Table 5–2 Printer Preferences Tab Window

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

5.1.3 Deleting a User

To delete a user:

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

5.2 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 the user to use the Application Console, and this user also belongs to a user group that permits the user to access only the Console and Applications Manager, the user has 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](#)

5.2.1 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 displays 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 5–3](#) for field value descriptions.
5. Choose .

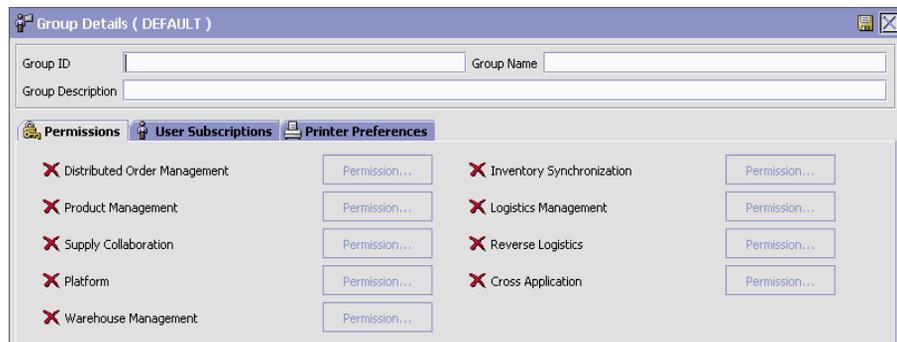


Table 5–3 Group Details Window

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.
Password Policy	Select the password policy you want to associate with the user group, from the drop-down list. For additional information about password policies, see the <i>Selling and Fulfillment Foundation: Password Policy Management Guide</i> .

To modify a user group:

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

You can also use the Group Details window for:

- [Administering User Group Permissions](#)
- [Setting Inner Panels as Read-Only](#)
- [Setting Up Permissions for Interoperability Servlet](#)
- [Setting Up Permissions for APIs](#)
- [Viewing the Users Subscribed to a User Group](#)
- [Setting Up Printer Preferences for a User Group](#)

5.2.1.1 Administering User Group Permissions

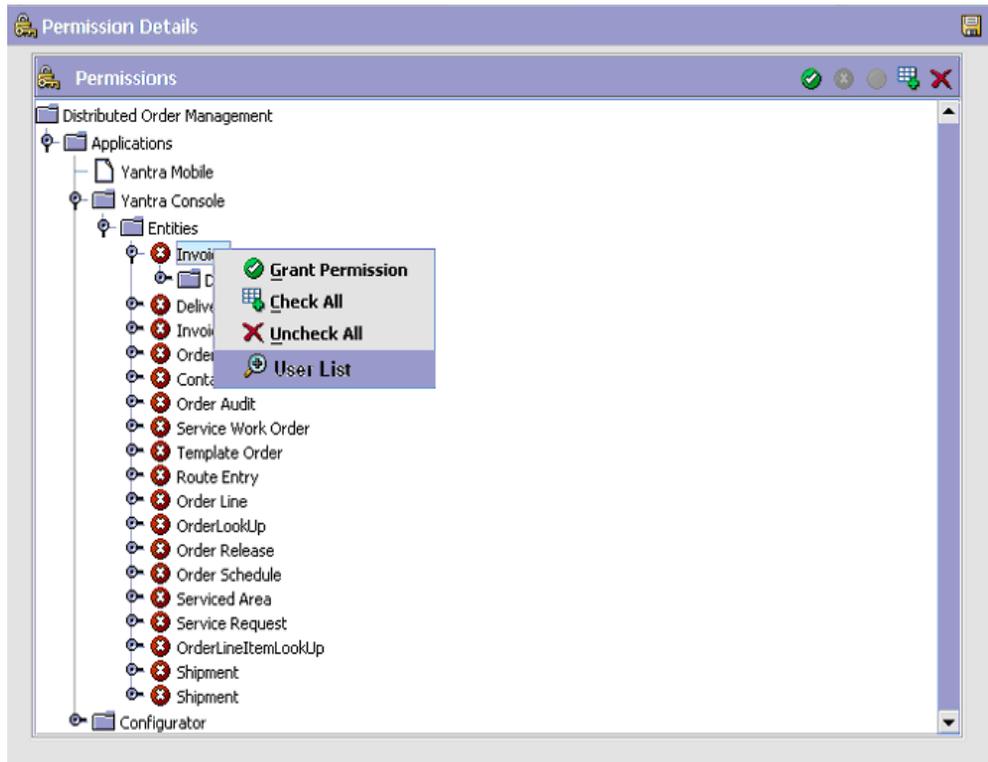
You can administer the permissions that a user group has throughout the Console and Applications Manager applications. You can allow or disallow permissions for an entire module or on a screen-by-screen or function-by-function basis. These permissions apply to all of the users in the user group.

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 displays.



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 do not see the System Management Console menu option in the Application Console.

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.

Note: The permissions tree displays the pricing screens and functions for both the new and old pricing functionalities. If you are using the new pricing functionality, permissions should be assigned to the new pricing functions. If you are using the old pricing functionality, permissions should be assigned to the old pricing screens. For more information about enabling the old and new pricing functionality, see [Section 6.3, "Defining Installation Rules"](#).

5. If you are configuring permissions for a group that has access to the Application Console, choose the Cross Application Permission button and expand the Application > Selling and Fulfillment Foundation 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, credit card name, credit card expiration date, stored value card number, customer account number or primary payment reference is displayed or masked in the Application Console.

If Selling and Fulfillment Foundation is configured to encrypt primary payment attributes, and the Display Decrypted Primary Payment Attributes permission is granted, the Application Console determines whether to call the `getDecryptedString` API to decrypt and display sensitive payment information.

For more information about enabling database encryption for primary payment attributes, see [Section 6.3, "Defining Installation Rules"](#).

- To grant the Application Console the ability to make modifications to 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 is 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 Selling and Fulfillment Foundation Console displays this field as editable, with a blue background.

- To grant the Application Console 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 Application Console, choose the Cross Application Permission button and expand the Application > Selling and Fulfillment Foundation 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.

5.2.1.2 Setting Inner Panels as Read-Only

Selling and Fulfillment Foundation enables you to mark inner panels as read-only. This ensures that a specified user group cannot modify or update any of the data visible to them. When an inner panel is marked as read-only, all the controls and its values in the page are visible but are disabled.

There are two ways to make an inner panel read-only:

- Set the read-only attribute for the resource in the RESOURCE_PERMISSION.xml file.

If the inner panel needs to be marked as read-only for an user, update the RESOURCE_PERMISSION_CONSOLE.xml file in the factory setup with the READ-ONLY attribute.

The following example shows the RESOURCE-PERMISSION.xml file with changes:

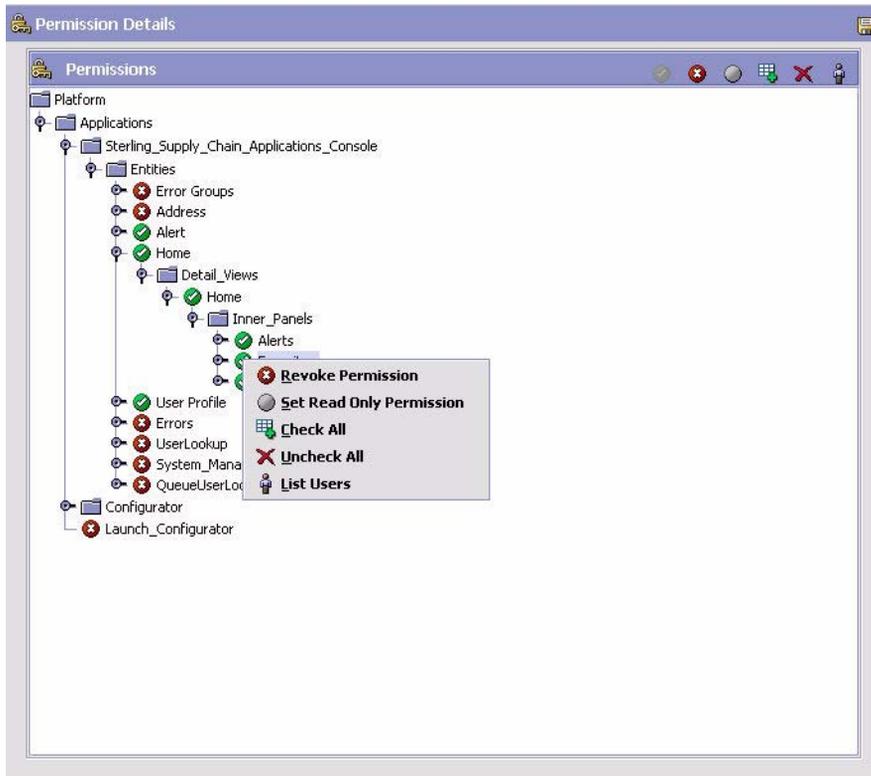
```
<ResourcePermission ActivateFlag="Y" Createprogid="SYSTEM"
Createuserid="SYSTEM" Modifyprogid="SYSTEM" Modifyuserid="SYSTEM"
ResourceKey="YEMS012" ResourcePermissionKey="SYS_YEMS012"
UsergroupKey="SYSTEM" READ-ONLY-FLAG="True"/>
```

In this example, the database table YFS-RESOURCE-PERMISSION will be updated for the user SYSTEM and resource YEMS012.

- Use the Applications Manager to apply security permissions and mark an inner panel as read-only.

1. Log in to the Applications Manager. Select **Applications > Application Platform > Security > Groups**.
2. Select a Group or create new Group. Click **Permission** next to the application you want to manage.

All of the entities for the application open in tree form. You can double click an entity to see its views and inner panels.



3. Right-click the selected inner panel to set or unset the inner panel as read-only.

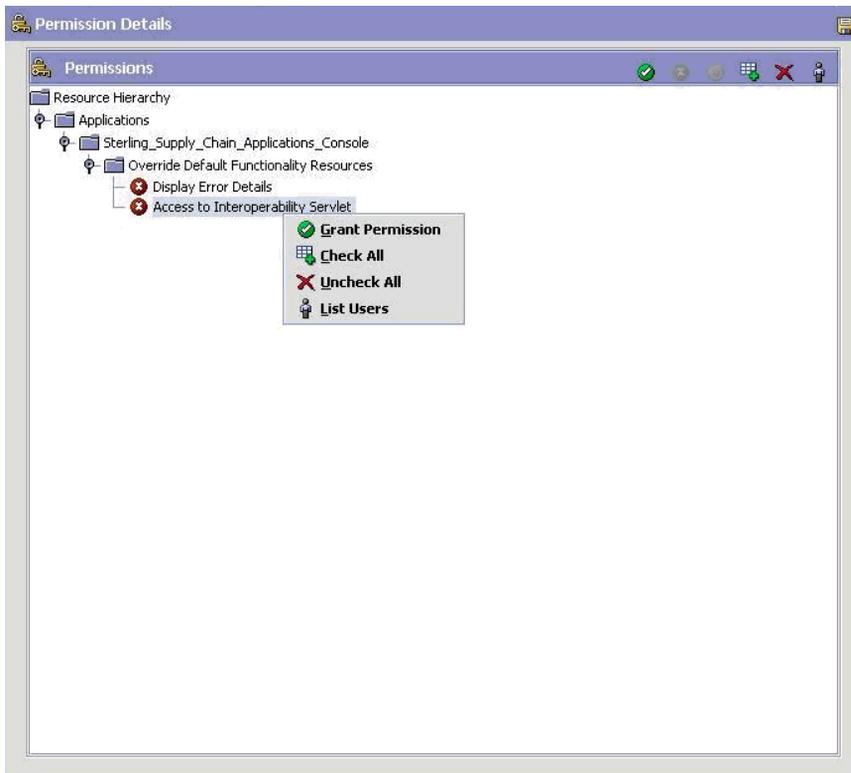
5.2.1.3 Setting Up Permissions for Interoperability Servlet

Selling and Fulfillment Foundation enables you to set up permissions for access to Interoperability Servlet. This ensures that a specified user group has permissions to access the Interoperability Servlet.

To set up security permissions for Interoperability Servlet using the Applications Manager:

1. Log in to the Applications Manager. Select **Applications > Application Platform > Security > Groups**.
2. Select a Group or create new Group. Click **Permission** next to the application you want to manage.

All of the entities for the application open in tree form. You can double click an entity to set Interoperability Servlet permissions for its elements.



3. Right-click the Access to Interoperability Servlet to grant permissions.

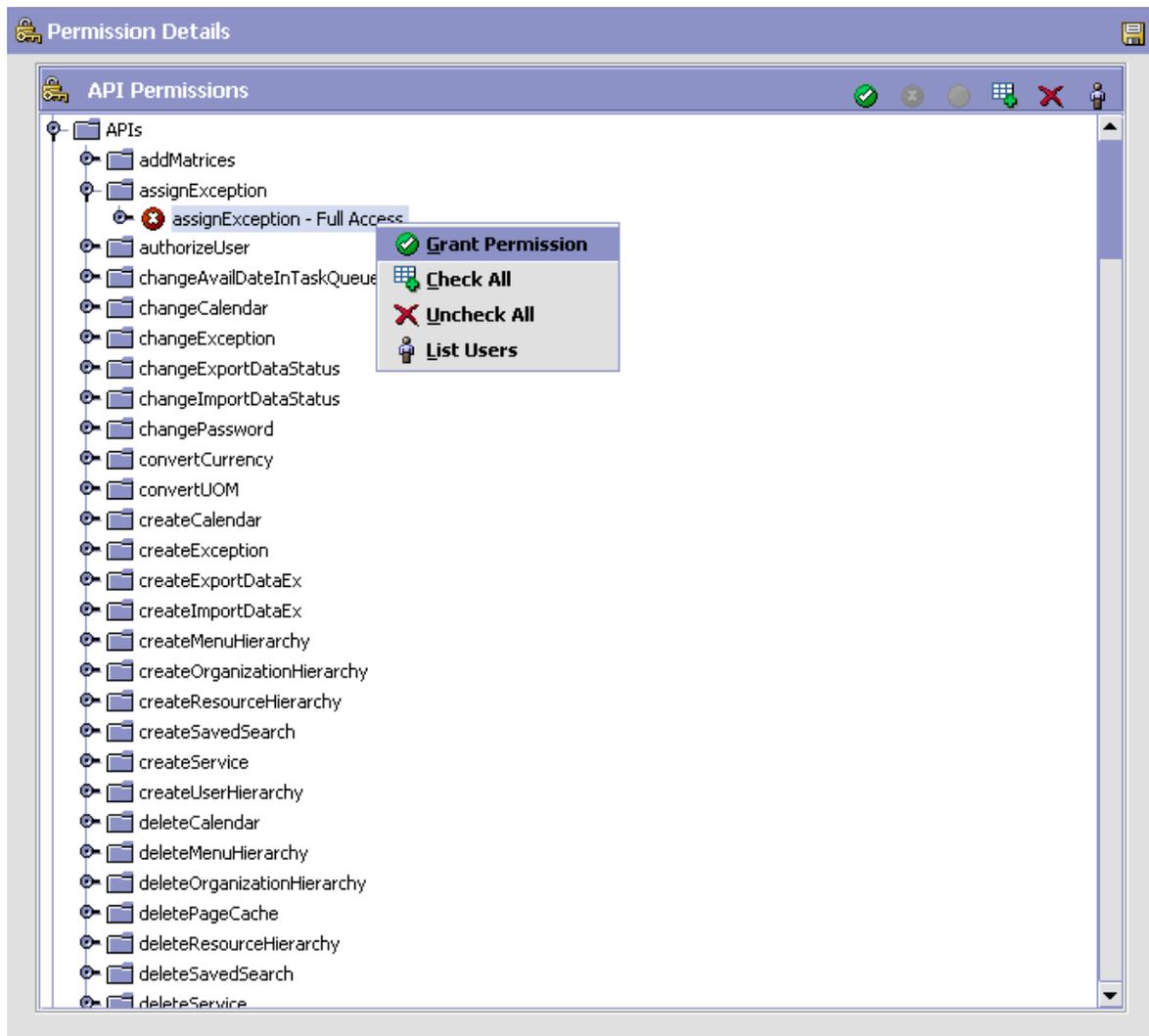
5.2.1.4 Setting Up Permissions for APIs

Selling and Fulfillment Foundation enables you to set up permissions for access to APIs for the authorization check. You must define access to API resources to control what can be accessed by users when calling an API.

To set up security permissions for APIs using the Applications Manager:

1. Log in to the Applications Manager. Select **Applications** > **Application Platform** > **Security** > **Groups**.
2. Select a Group or create new Group. Click **Permission** next to the API Security. The Api Permissions pop-up window displays.

You can double click on APIs or Services branch. All of the APIs and Services for the Selling and Fulfillment FoundationApplication open in tree form.



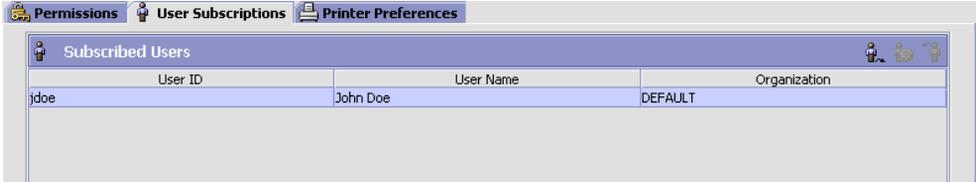
3. Right-click the individual API or Service to grant permissions.
4. At the upper right of the API Permissions screen, click  or  to grant or revoke permission to the user for that API or Service. By default, permission is revoked for all APIs or Services. Click  to

list all of the users who have been granted permission for an API or Service.

5. When you are finished granting and revoking permissions, save the permission record by clicking .

5.2.1.5 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

5.2.1.6 Setting Up Printer Preferences for a User Group

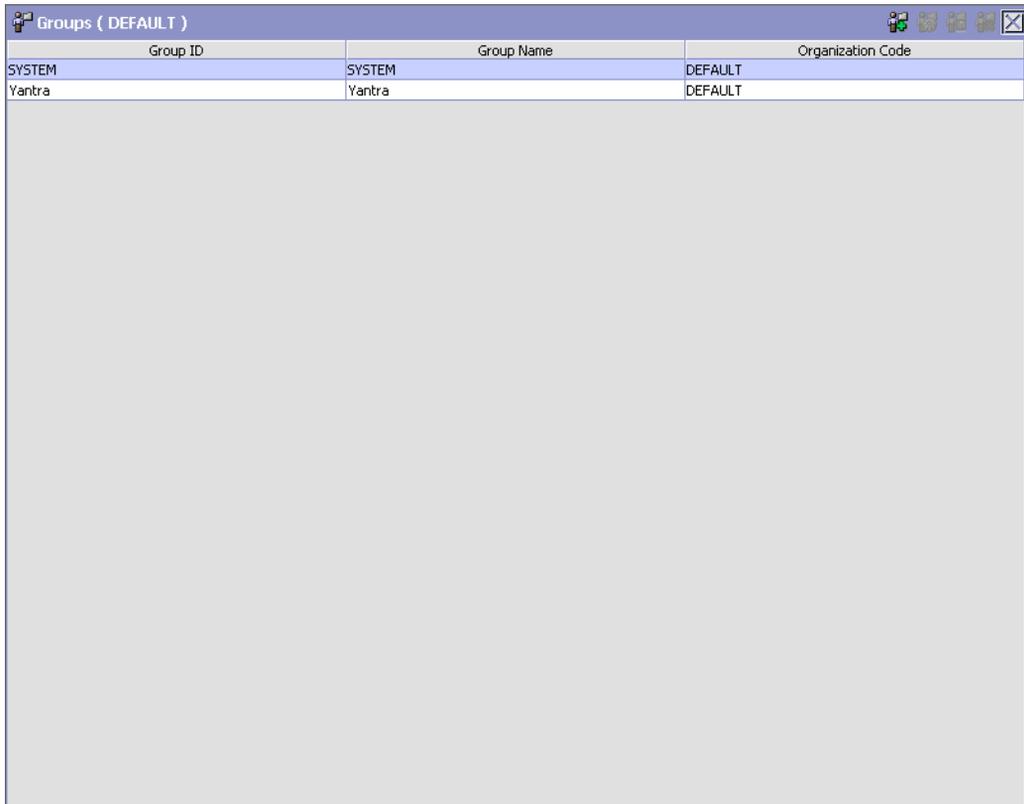
User Printer Preferences configures printers that are associated with a group of users. This preference is used to determine the printer to use when a user prints a document.

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 > Application Platform. The Application Platform tree displays in the side panel.
2. From the tree in the application rules side panel, choose Security > Groups. The Groups window displays 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 displays.
5. In the Group Details window, choose the Printer Preferences tab. The Printer Preferences tab window displays.
6. Enter the information in the applicable fields. Refer [Table 5–4](#) for field value descriptions.
7. Choose .

For more information about Setting Up a Group (Creating, Modifying, or Deleting a Group), [Section 5.2, "Defining User Groups"](#).

Group Details (DEFAULT)

Group ID: Yantra Group Name: Yantra

Group Description: Yantra

Permissions User Subscriptions Printer Preferences

Printer Association

Printer ID
*

Results 0 Of 0

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 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.

5.2.2 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 displays in the work area.
2. Select the applicable user group and choose .

5.3 Defining Teams

A **Team** is a collection of users who have common data access requirements. Teams can have access to specific document types, Enterprises, ship nodes, and customers. Teams can be assigned to specific customers.

Creating a team is an optional process. If a user is not associated with a team, that user is considered to have the least restrictive access, or default access to customer orders and information. By defining a team, you can further restrict the access to any Enterprises, document types, or participating ship nodes that are a sub-set of the default access list. For more information about teams and how you can use them in Selling and Fulfillment Foundation, see the *Selling and Fulfillment Foundation: Product Concepts Guide*.

Note: Clearing of the database cache is required to take into effect the changes made to this configuration.

You can use the Teams branch for:

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

5.3.1 Creating a Team

To create a team:

1. From the tree in the application rules side panel, choose Security > Teams. The Teams window displays in the work area.
2. Choose . The Team Details window displays.

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

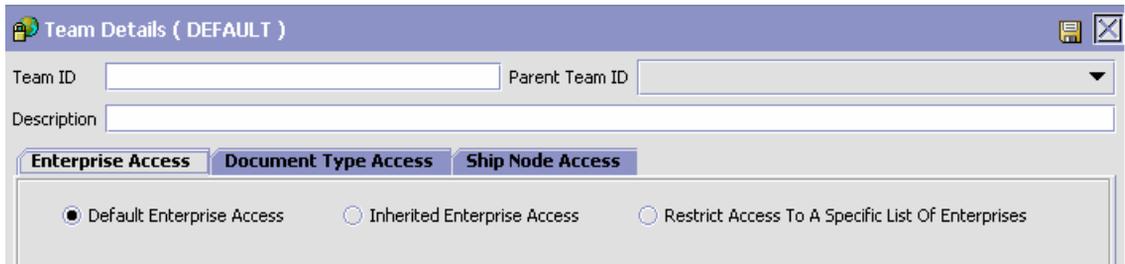


Table 5–5 Team Details Window

Field	Description
Team ID	Enter the team name.
Parent Team ID	Select the name of the team that is the parent team for this team. If this team is part of a multi-level hierarchical team structure, this must be the name of a team that is in the next higher level within the structure.
Description	Enter a brief description of the user team.
Enterprise Access	
Default Enterprise Access	Select Default Enterprise Access to restrict users who belong to this team to be able to only view the Enterprises belonging to the default Enterprise access list.
Inherited Enterprise Access	Select Inherited Enterprise Access to restrict users who belong to this team to be able to only view the Enterprises based on the access configuration for the parent team.

Table 5–5 Team Details Window

Field	Description
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 team 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 always includes the primary Enterprise of the organization you are configuring the team for.</p>
Document Type Access	
Default Document Type Access	<p>Select Default Document Type Access to allow this team 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.</p>
Inherited Document Type Access	<p>Select Inherited Document Type Access to restrict users who belong to this team to be able to only view the document types based on the access configuration for the parent team.</p>

Table 5–5 Team Details Window

Field	Description
Restrict Access 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 team 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 displays displaying all of the accessible document types available for the business application. Select the Accessible check boxes for the applicable document types.</p> <p>Note: At least one document type must be included in a data security group for each business application.</p>
Ship Node Access	
All Nodes	<p>Select this radio button to allow the team to access all the ship nodes.</p> <p>Note: This radio button is not available for an organization whose role is Node.</p>
User's Node	<p>If the user is a node user, selecting this radio button enables the team to access the node to which the node user belong.</p> <p>If the user is an enterprise user, selecting this radio button allows the team to access all the nodes whose parent organization is the user's enterprise, an enterprise that is hierarchically lower than the user's enterprise, or both.</p>

Table 5–5 Team Details Window

Field	Description
Restrict Access To Specific Nodes	<p>Select this radio button to restrict the team’s access to only specific nodes, and save the setting.</p> <p>On saving,  is displayed in the Accessible ShipNode panel. Click , and search for the nodes from the ShipNode Search dialog box.</p> <p>Note: The ShipNode Search dialog box displays only those nodes that are accessible to the administrator creating the team.</p> <p>Select the node and click  to make this node accessible to the team. The selected node will be displayed in the Accessible ShipNode panel.</p> <p>To remove a node from the Accessible ShipNode panel, select the node and click . A confirmation dialog box is displayed. Select OK to confirm the deletion of the node.</p>
Nodes Accessible To Team Creator	<p>Select this radio button to allow the team to access all the nodes that are accessible to the creator of the team.</p> <p>Notes:</p> <ul style="list-style-type: none"> • If the creator of the team is a HUB user (Organization = DEFAULT), the team can access all the ship nodes. • If the creator’s access to certain nodes changes, the team’s access to those nodes will also change automatically. Also, the team’s access to the nodes ceases when the creator is deleted from the Applications Manager.

5.3.2 Modifying a Team

To modify a team:

1. From the tree in the application rules side panel, choose Security > Teams. The Teams window displays in the work area.
2. Select the applicable team and choose . The Team Details window displays.

3. Make revisions to the applicable fields. Refer to [Table 5–5](#) for field value descriptions.
4. Choose .

5.3.3 Deleting a Team

To delete a team:

1. From the tree in the application rules side panel, choose Security > Teams. The Teams window displays in the work area.
2. Select the applicable team and choose .

5.4 Defining Data Access Policies

Because all applications have different filtering requirements for data access security, Selling and Fulfillment Foundation offers a variety of data access policies that you can apply.

You can define data access policies for Enterprise, Buyer, Seller, and Node users so that you can control access to data contained in specific document types, enterprises, and ship nodes. Data access policies apply to the following access modes:

- [Enterprise User Access](#)
- [Buyer User Access](#)
- [Seller User Access](#)
- [Node User Access](#)

You can define these access policies as follows:

1. From the tree in the application rules side panel, choose Security > Data Access Policy Rules. This window displays in the work area.
2. Choose among the Enterprise, Buyer, Seller, and Node User Access tabs displayed.

The following sections describe how to configure these access modes. For more information about how the Data Access Policies can be used, see the *Selling and Fulfillment Foundation: Product Concepts Guide*.

Notes:

- Clearing of the database cache is required to take into effect the changes made to this configuration.
 - The Selling and Fulfillment Foundation Applications Manager does not load configuration data based on the Data Access Policies described in this section.
-
-

5.4.1 Enterprise User Access

You can configure access for an Enterprise user, such as an internal storefront administrator, in the Data Access Policy Enterprise User screen.

1. Choose the Enterprise User tab.
2. Enter information in the applicable fields. Refer to [Table 5–6](#) for field value descriptions.



Table 5–6 Enterprise User Access Policy Rules

Field	Description
Enterprise Access	
Users Have Access To Their Enterprise Only	Users can have access to data belonging to only their enterprise.
Users Have Access To All Enterprises Assigned To Their Team	Users can have data access to all enterprises assigned to their team.
Customer Access	
Users Have Access To All Customers	Users can have access to all customers in the hierarchy.
Users Have Access To Selected Customers According To Team Configuration	Users can have access to a customer only as defined by rules for customer hierarchy access.

Table 5–6 Enterprise User Access Policy Rules

Field	Description
Customer Hierarchy Access	
Users Have Access To Customers And Their Child Customers	Users can have access to customers and the child customers.
Users Have Access To Their Customers	Users can have access to customers who are directly assigned. (This requires manual user-to-customer assignment to be configured in the team definition.)
Supervisor Access	
Supervisors Have Access To Customers Assigned To Their Team	Supervisors can have access to all the customers that their team has access to.
Supervisors Have Access To Customers Assigned To Their Team And Child Team	Supervisors can have access to all the customers assigned to their team and all child teams.

5.4.2 Buyer User Access

You can configure access for a buyer user, such as a customer buyer, in the Data Access Policy Buyer User tab. Refer to [Table 5–7](#) for field value descriptions.

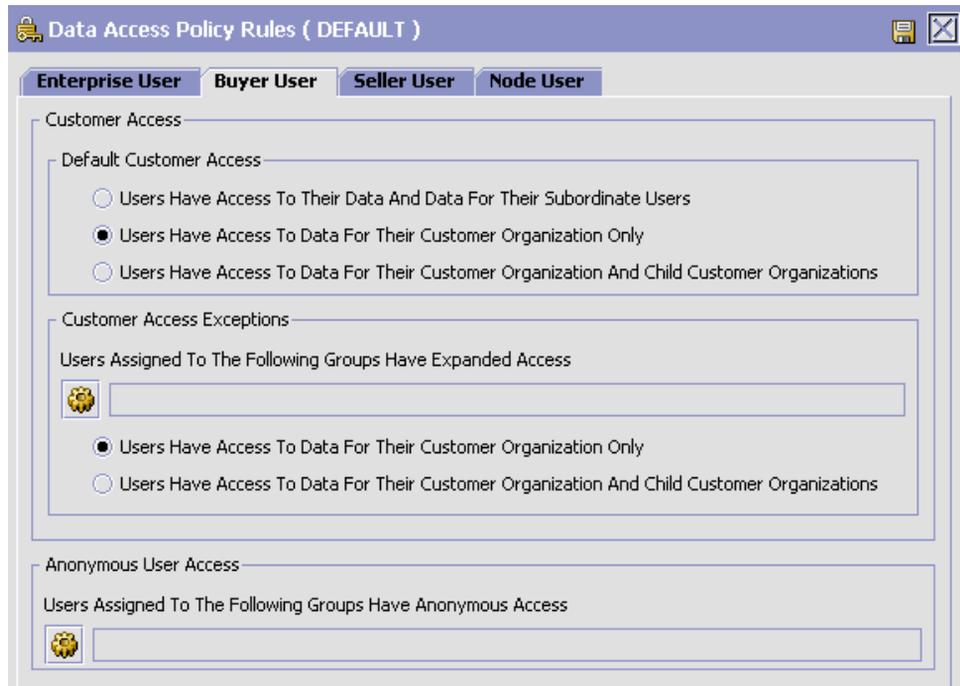


Table 5–7 Buyer User Access Policy Rules

Field	Description
Customer Access	
Default Customer Access	
Users Have Access To Their Data And Data For Subordinate Users	Buyers can have access to their own data and the data of subordinate users.
Users Have Access To Data For Their Customer Organization Only	Buyers can have access to data only for their customer organization.
Users Have Access To Data For Their Customer Organization And Child Customer Organizations	Buyers have access to data for their customer organization and child customer organizations.

Table 5–7 Buyer User Access Policy Rules

Field	Description
Customer Access Exceptions	
Users Assigned To The Following Groups Have Expanded Access	<p>Select whether buyers can have expanded data access based on their subscription to a group.</p> <p>Choose  to display Group Subscriptions.</p>
Group Subscriptions	
Available	<p>A list of the available user groups.</p> <p>To subscribe a user to a user group, select the applicable user group and choose .</p>
Subscribed	<p>A list of the user groups to which the user is subscribed.</p> <p>To remove a user from a user group, select the applicable user group and choose .</p>
Users Have Access To Data For Their Customer Organization Only	Buyers can have access to data only for their customer organization.
Users Have Access To Data For Their Customer Organization And Child Customer Organizations	Buyers can have access to data only for their customer organization and child organizations.
Anonymous User Access	
Users Assigned To The Following Groups Have Anonymous Access	<p>Select whether users can have anonymous data access based on their subscription to a group.</p> <p>Choose  to display Group Subscriptions.</p>
Available	<p>A list of the available user groups.</p> <p>To subscribe a user to a user group, select the applicable user group and choose .</p>
Subscribed	<p>A list of the user groups to which the user is subscribed.</p> <p>To remove a user from a user group, select the applicable user group and choose .</p>

5.4.3 Seller User Access

For seller users, the following access rules exist. They are not configurable at this time. Refer to [Table 5–8](#) for field value descriptions.



Table 5–8 Seller User Access Policy Rules

Field	Description
Seller Access	
Users Have Access To Their Organization Only	Sellers can have access only to their organization.

5.4.4 Node User Access

You can configure node user access in the Data Access Policy Node User tab. Refer to [Table 5–9](#) for field value descriptions.



Table 5–9 Node User Access Policy Rules

Field	Description
Node Access	
Users Have Access To Their Node Only	Users have access only to their node.
Users Have Access To Nodes Assigned To Their Team	Users can have access to any nodes assigned to their team.

5.5 Configuring API Security

You must define access to API resources to control what can be accessed by users when calling an API.

When calling an API, you must pass through the following two levels of security:

1. Authentication with a user ID, a certificate or both. The login API is called before any other API is called.
2. Authorization, which verifies which resources you can access.

For more information about API security, refer to the *Selling and Fulfillment Foundation: Customizing APIs Guide*.

Note: If you're running Selling and Fulfillment Foundation components as Web services with API security enabled, you must expose the Login API as a Web service. Refer to the *Selling and Fulfillment Foundation Installation Guide* for details about exposing APIs when preparing to build Web services.

Additionally, you must call the Login API, capture the security token that is generated at login, and then set the token as the "tokenId" in `YFSEnvironment`. For details about the `YFSEnvironment` interface, see the *Selling and Fulfillment Foundation Javadocs*.

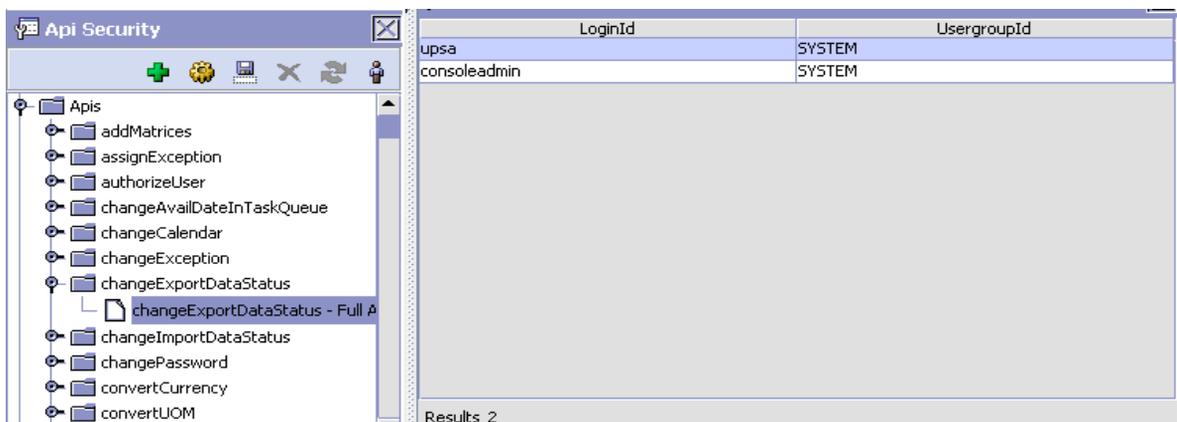
You can use the API Security branch for:

- [Viewing API Security Resources](#)
- [Creating an API Security Resource](#)
- [Creating an API Security Resource](#)
- [Modifying an API Security Resource:](#)

5.5.1 Viewing API Security Resources

To view the list of existing API Security Resources:

1. From the tree in the application rules side panel, choose Security > API Security. The API Security Editor window displays.
2. Expand the APIs or Services branch in Selling and Fulfillment Foundation Select an API or Service. By default, Full Access option is available for an API or Service. Click  to list all of the users who have been granted permission for an API or Service.

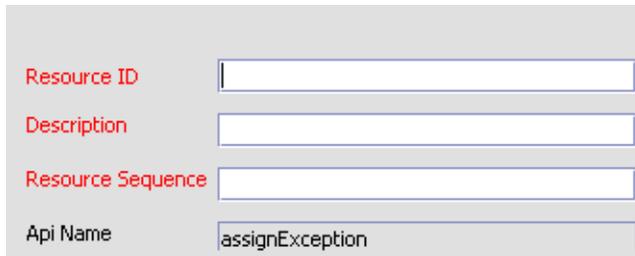


5.5.2 Creating an API Security Resource

To create an API Security Resource:

1. From the tree in the application rules side panel, choose Security > API Security. The API Security Editor window displays.

2. Expand the APIs or Services branch. Selling and Fulfillment FoundationSelect an API for which you want to create new API Security Resource and click . The Create Api Security Resource window displays.



The screenshot shows a form with four input fields. The labels are in red text: 'Resource ID', 'Description', 'Resource Sequence', and 'Api Name'. The 'Api Name' field contains the text 'assignException'.

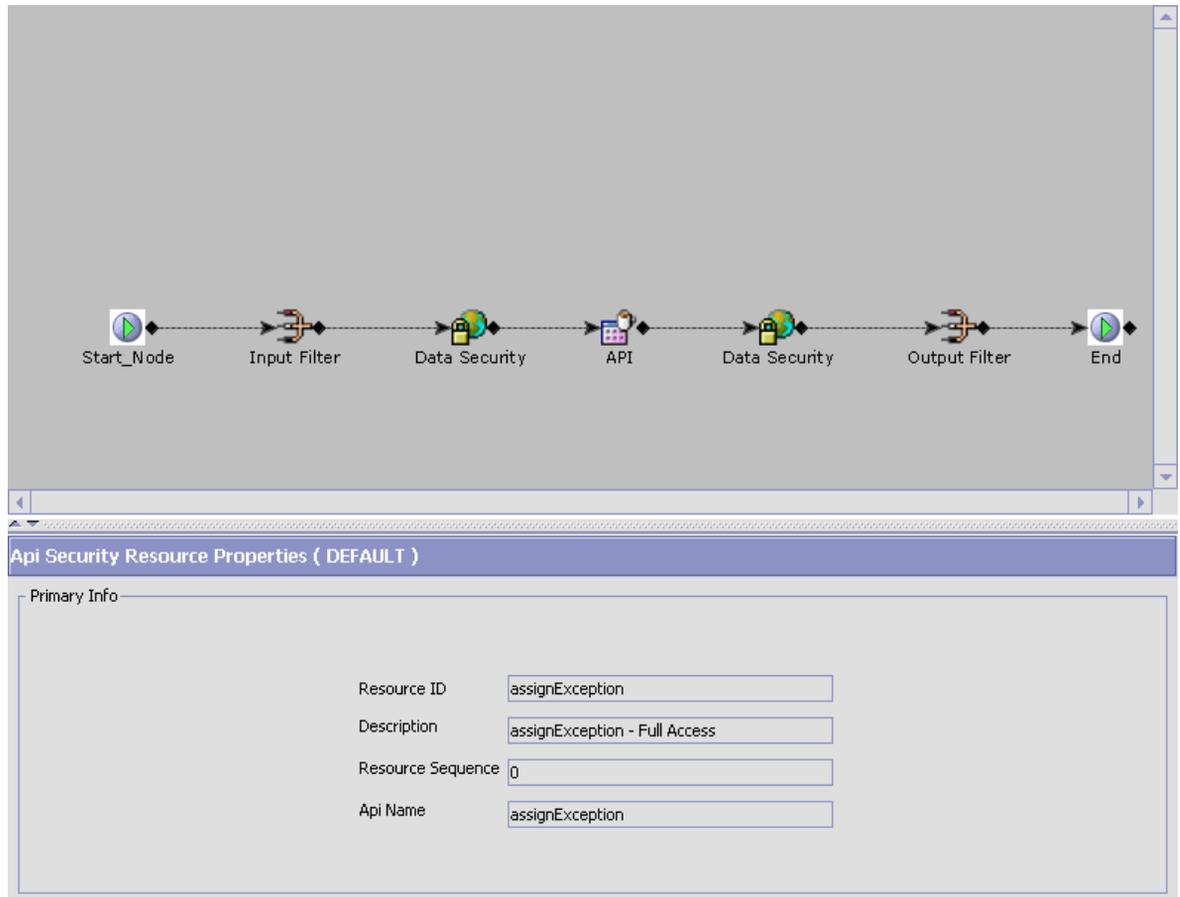
3. In Resource ID, enter the unique identifier for the API resource.
4. In Description, enter a brief description of the API resource.
5. In Resource Sequence, enter order in which this API resource needs to be invoked. The output of one API resource may be used as input to another.
6. Choose .

5.5.3 Modifying an API Security Resource:

Once an API Security Resource has been defined, it can be modified.

To modify an API Security Resource:

1. From the tree in the application rules side panel, choose Security > API Security. The API Security Editor window displays.
2. Expand the APIs or Services branch in Selling and Fulfillment FoundationSelect the individual API or Service branch which you want to modify and choose . The API Security Resource Details window and API Security Resource Properties window displays.



3. The API Security Resource Details window displays the various components associated by default with each API Security Resource. If none of the components is selected in the API Security Resource Details window, the API Security Resource Properties window will display the read only information relates to the selected API Security Resource.
4. Once you select a particular component, the details of that component will appear in the lower property window.

5. You can configure the following components associated with an API Security Resource:

Note: When you select Start or End component, the property window This property screen will simply display the default input apisecurity file and output apisecurity file respectively in a text area for quick reference. It cannot be edited, but you can copy the text.

Input Filter

The Input Filter component provides a mean to edit the API filter input XML.



Click  to import the default api filter input XML. Click  to load the api filter input XML from a file. Click  to load the api filter input XML from string.

To manipulate the Input Filter XML element structure, select the node and click  to add a new child element in the XML tree, and click  to add a new attribute for an element in the XML tree.

Select the appropriate attribute of an element whose value you want to set or modify and click .

If you want to delete an element or attribute from the XML tree, select the appropriate element or attribute and click .

If you don't want an attribute to be processed during API call. select the appropriate attribute and click  and if you want to change the status of an attribute from Unselect to Select, click .

Data Security

The Data Security component allows you to define a list of XPath expressions that identify organizations to validate. Click  to add a new XPath expression.

Note: Same configuration will be used for both input as well as output API security.

API

The API component is used for the API name. It is non-editable.

Output Filter

The Output Filter component is similar to the Input Filter component except it provides a mean to edit the API filter output XML.

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](#)
- [Defining Application Version](#)

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 the Selling and Fulfillment Foundation.

Table 6–1 System-Defined Purge Rules

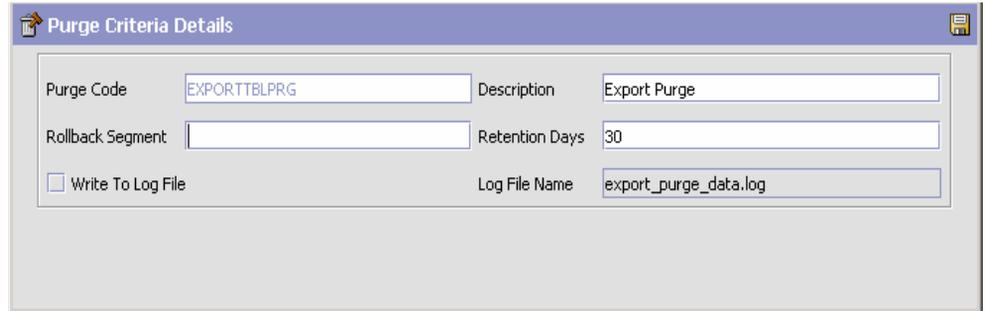
Rule	Description	Retention Days
CAPACITYPRG	Purges capacity data.	30
EXPORTBLPRG	Purges data from export tables that are used for publishing data to external systems. This is a Hub level purge.	30
IMPORTBLPRG	Purges data from the import tables. This is a Hub level purge.	30
INBOXPRG	Purges the Alert Console messages from the user's inbox. This is a Hub level purge.	30
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
MANIFESTPRG	Purges manifest information.	30
PERSONINFOHISTPRG	Purges historical customer information.	30
PERSONINFOPRG	Purges customer information and moves it to a history table.	30
PRICELISTPRG	Purges price lists.	30
REPROCESSPRG	Purges any reprocessed information. This is a Hub level purge.	30
STATBLPRG	Purges statistical information. This is a Hub level purge.	30
USERACTAUDITPRG	Purges all user activity audit data from the system.	30
USERACTIVITYPRG	Purges all user activity data.	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 displays in the work area.

2. Select the applicable purge criteria rule that and choose . The Purge Criteria Details pop-up window displays.



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 to log system messages for the status of the purge. The log can be backed up and used as a journal at a later date.
Log File Name	<p>Enter a log file name. The log file is created in the directory specified in the <code>yfs.purge.path</code> property. If this is not passed, it defaults to the value specified in the <code>yfs.properties</code> file. If a variable is introduced, then the <code>yfs.purge.path</code> is ignored.</p> <p>To modify this property, add an entry for it in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file. For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p> <p>For more information about using variables for the log file directory, see the <i>Selling and Fulfillment Foundation: Extending Transactions Guide</i>.</p> <p>For information about file name limitations related to internationalization, see the <i>Selling and Fulfillment Foundation: Localization Guide</i>.</p>

6.2 Defining User Exit Implementations

User exits are created to enable business logic extensions to the Selling and Fulfillment Foundation transactions. Within the Selling and Fulfillment Foundation transactions, code exists to invoke user exits so that you may plug-in custom logic. Since these are pre-defined by Selling and Fulfillment Foundation, 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 the Selling and Fulfillment Foundation transactions can invoke them to perform the necessary logic at runtime. This chapter explains how to configure user exit implementations within Selling and Fulfillment Foundation.

Note: If you do not require the Selling and Fulfillment Foundation transaction extension, you do not need to implement user exits. If a user exit is not configured, Selling and Fulfillment Foundation runs 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"](#). 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, the Selling and Fulfillment Foundation 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 have to be kept in mind when you are using User Exits within the Selling and Fulfillment Foundation API:

- User Exits are structured to return specific information and their usage must be restricted to such purposes.

Inheritance

You can configure inheritance for resources such as user exits (and their events) and templates at the Enterprise level. For example, instead of

defining a whole new set of resource configurations, Enterprise "B" may choose to inherit the resource configuration from Enterprise "A".

You can configure inheritance at the Enterprise level for:

- . User Exits
- . Templates

Inheritance for User Exits

You can inherit user exit implementations at the Enterprise level.

If a user exit implementation exists for the Enterprise, the system returns that user exit implementation. But if the user exit implementation is not defined for that Enterprise, the system checks to see if it inherits configuration from some other enterprise and uses the inheritance hierarchy to search for the correct user exit implementation. If no user exit implementation is found, the Hub's implementation of the user exit for that Enterprise is used.

Inheritance for Events

You can inherit event implementations at the Enterprise level.

<placeholder> (Need to consume from Application Platform.)

Inheritance for Templates

You can inherit the user exit and event templates at the Enterprise level by specifying the resource identifier for an Enterprise. If the resource identifier is not specified, it is obtained through the inheritance hierarchy.

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 displays in the work area.

User Exit List (DEFAULT)				
User Exit Name	Can Override For Document Type	Can Override For Enterprise	Can Attach Service	User Exit Implemented
com.yantra.yfs.japi.ue.YFS...				
com.yantra.yfs.japi.ue.YFS...				
com.yantra.yfs.japi.ue.YFS...				
com.yantra.ycp.japi.ue.YC...			✓	
com.yantra.yfs.japi.ue.YFS...				
com.yantra.ycp.japi.ue.YC...			✓	
com.yantra.ycp.japi.ue.YC...			✓	
com.yantra.ycp.japi.ue.YC...			✓	
com.yantra.yfs.japi.ue.YFS...				
com.yantra.ycs.japi.ue.YC...				
com.yantra.wms.japi.ue.W...			✓	
com.yantra.wms.japi.ue.W...			✓	
com.yantra.wms.japi.ue.W...		✓	✓	
com.yantra.vas.japi.ue.VA...		✓	✓	
com.yantra.vas.japi.ue.VA...		✓	✓	
com.yantra.vas.japi.ue.VA...		✓	✓	
com.yantra.vas.japi.ue.VA...		✓	✓	
com.yantra.vas.japi.ue.VA...		✓	✓	
com.yantra.vas.japi.ue.VA...		✓	✓	
com.yantra.ycs.japi.ue.YC...				
com.yantra.wms.japi.ue.W...		✓	✓	
com.yantra.wms.japi.ue.W...		✓	✓	
com.yantra.wms.japi.ue.W...		✓	✓	
com.yantra.wms.japi.ue.W...		✓	✓	
com.yantra.wms.japi.ue.W...		✓	✓	
com.yantra.wms.japi.ue.W...		✓	✓	
com.yantra.wms.japi.ue.W...		✓	✓	
com.yantra.wms.japi.ue.W...		✓	✓	

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'. If the User Exit can be overridden by an Enterprise, the 'Can Override for Enterprise' column displays 'Y'.

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

User Exit Name:

This User Exit can be overridden for Document Types

This User Exit Can Be Designed As a Service

This User Exit can be overridden for an Enterprise

Document Type	Use Service	Service Name	Java Class Name
---------------	-------------	--------------	-----------------

Results 0 of 0

Document Type	Use Service	Service Name	Java Class Name	Enterprise Code
---------------	-------------	--------------	-----------------	-----------------

The following three fields at the top of the screen are informational and read-only:

- This User Exit can be overridden for Document Types
- This User Exit can be overridden for an Enterprise
- This User Exit Can Be Designed As a Service

These fields reflect the options selected on the previous screen. The Inherited User Exit Implementation panel is also informational and read-only. This panel displays the list of user exits that are inherited by the current Enterprise from other Enterprises in its hierarchy. For example, consider three Enterprises, E1, E2, and E3, that have the following hierarchical relationship:

- Enterprise E1 inherits configurations from Enterprise E2.
- Enterprise E2 inherits configurations from the Enterprise E3.

Because of this hierarchy, Enterprise E1 can view and override all the configurations that are defined for Enterprise E2 and E3. Enterprise E1 can also view the configurations defined at the Hub level.

You can add values to the table in the User Exit Implementation List panel.

- To add values to the User Exit Implementation List table, choose  in the User Exit Implementation panel. The User Exit Implementation Details displays. Enter information into the applicable fields. Refer to [Table 6–3](#) for field value descriptions.

The screenshot shows a configuration window titled "User Exit Implementation Details". It contains several input fields and checkboxes:

- Document Type:** A dropdown menu.
- Implement As a Service:** A radio button (unselected) and a dropdown menu.
- Implement As a Java Class:** A radio button (selected) and a text input field.
- Requires Backward Compatibility:** A checkbox (unchecked) and a dropdown menu.
- Restrict Number Of Calls:** A checkbox (unchecked).
- 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 for notes.

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.

Table 6–3 User Exit Implementation Details Fields

Field	Description
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" .
Java Class (if selected Implement as Java Class)	If you selected Implement As a Java Class, enter the Java class as it displays 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 Selling and Fulfillment Foundation 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 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 cause an error.
Wait Time (seconds)	Time for which the user exit call waits in queue. If the wait time exceeds the configured wait time, an exception is thrown.
User Exit Implementation Notes	Enter any additional information regarding user exit implementation.

6.3 Defining Installation Rules

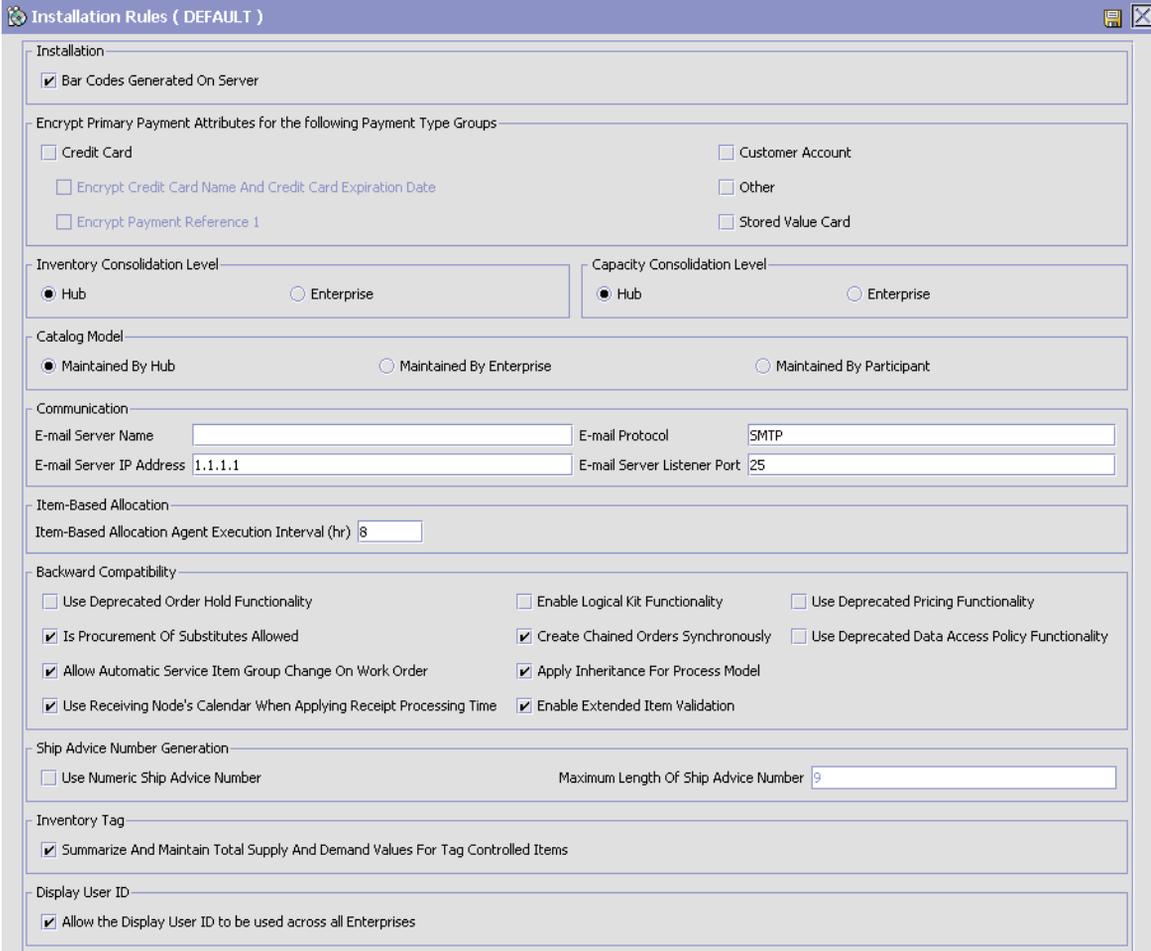
You can set up rules that need to be defined when the Hub installs the application.

To set up installation rules:

1. From the tree in the application rules side panel, choose System Administration > Installation Rules. The Installation Rules window displays in the work area.

2. Enter information in the applicable fields. Refer to [Table 7–16](#) for field value descriptions.
3. Choose .

Figure 6–1 Installation Rules



Installation Rules (DEFAULT)

Installation

Bar Codes Generated On Server

Encrypt Primary Payment Attributes for the following Payment Type Groups

Credit Card Customer Account

Encrypt Credit Card Name And Credit Card Expiration Date Other

Encrypt Payment Reference 1 Stored Value Card

Inventory Consolidation Level

Hub Enterprise

Capacity Consolidation Level

Hub Enterprise

Catalog Model

Maintained By Hub Maintained By Enterprise Maintained By Participant

Communication

E-mail Server Name E-mail Protocol

E-mail Server IP Address E-mail Server Listener Port

Item-Based Allocation

Item-Based Allocation Agent Execution Interval (hr)

Backward Compatibility

Use Deprecated Order Hold Functionality Enable Logical Kit Functionality Use Deprecated Pricing Functionality

Is Procurement Of Substitutes Allowed Create Chained Orders Synchronously Use Deprecated Data Access Policy Functionality

Allow Automatic Service Item Group Change On Work Order Apply Inheritance For Process Model

Use Receiving Node's Calendar When Applying Receipt Processing Time Enable Extended Item Validation

Ship Advice Number Generation

Use Numeric Ship Advice Number Maximum Length Of Ship Advice Number

Inventory Tag

Summarize And Maintain Total Supply And Demand Values For Tag Controlled Items

Display User ID

Allow the Display User ID to be used across all Enterprises

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	
Note: Sterling Commerce recommends that payment information entering the system is already tokenized instead of being encrypted by the following rules.	
Credit Card	If checked, Selling and Fulfillment Foundation automatically encrypts the credit card number in the database. Only users with the appropriate permissions can view decrypted credit card numbers. For more information about configuring permissions, see Section 5.2.1.1, "Administering User Group Permissions" .
Additionally Encrypt Credit Card Name And Credit Card Expiration Date	If checked, only the Selling and Fulfillment Foundation users with the appropriate permissions can view decrypted credit card numbers. For more information about configuring permissions, see Section 5.2.1.1, "Administering User Group Permissions" . Note: Credit Card must be checked first.
Stored Value Card	If checked, Selling and Fulfillment Foundation 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 about configuring permissions, see Section 5.2.1.1, "Administering User Group Permissions" .
Customer Account	If checked, Selling and Fulfillment Foundation 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 about configuring permissions, see Section 5.2.1.1, "Administering User Group Permissions" .

Table 6–4 Installation Rules Window

Field	Description
Other	<p>If checked, Selling and Fulfillment Foundation 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 about configuring permissions, see Section 5.2.1.1, "Administering User Group Permissions".</p>
Inventory Consolidation Level	
Hub	<p>Select the Hub option if you want product item IDs to remain unique across all organizations.</p> <p>Important: When an organization is created and this option is selected, the inventory organization is 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".</p>
Enterprise	<p>Select the Enterprise option to allow product item IDs to repeat across enterprises but still be distinguished within the inventory model. Also select the Enterprise option if you want to add this Enterprise organization to a colony different from the DEFAULT colony.</p> <p>Important: When creating an Enterprise organization and this option is selected, the inventory organization is 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".</p> <p>Note: If you choose Enterprise, an organization or Enterprise code must be specified when performing an inventory adjustment in the Consoles.</p>

Table 6–4 Installation Rules Window

Field	Description
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 is 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".</p>
Enterprise	<p>Select the Enterprise option to allow service item IDs to repeat across enterprises but still be distinguished within the inventory model. Also select the Enterprise option if you want to add this Enterprise organization to a colony different from the DEFAULT colony.</p> <p>Important: When creating an Enterprise organization and this option is selected, the inventory organization is 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".</p>
Catalog Model	
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 is 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".</p>

Table 6–4 Installation Rules Window

Field	Description
Maintained by Enterprise	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. Important: When an organization is created and this option is selected, the catalog organization is 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" .
Maintained by Participant	Select the Maintained by Participant option if you want each individual participant to maintain their own catalog. Important: When an organization is created and this option is selected, the catalog organization is 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" .
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.
Item Based Allocation	
Item Based Allocation Agent Execution Interval (hr)	This rule is used to indicate the interval during which the Item Based Allocation agent should not reprocess the triggers that were processed earlier in the YFS_IBA_TRIGGER table.

Table 6–4 Installation Rules Window

Field	Description
Backward Compatibility	
Use Deprecated Order Hold Functionality	<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 about the enhanced order hold functionality, see the <i>Sterling 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>
Enable Logical Kit Functionality	<p>Check this box if you want to enable logical kit functionality. By default this box is not checked as bundles is an enhanced feature of the logical kit.</p>

Table 6–4 Installation Rules Window

Field	Description
Is Procurement of Substitutes Allowed	<p>Check this box if you want to enable the procurement of substitutes.</p> <p>An order gets scheduled and not backordered, if inventory for the substitutes can be procured.</p>
Create Chained Orders Synchronously	<p>Check this box to create procurement and drop ship orders synchronously when scheduling.</p> <p>If this rule is enabled, chained orders are created only if the expected ship date on the child order is within the Advanced Notification Time of the child orders shipnode.</p>
Allow Automatic Service Item Group Change on Work Order	<p>Check this box to allow automatic changes in the Service Item Group when the Provided Service line is added or removed from the work order or when the Service Item Group of a work order is changed from one type of service to the other. For example, from a Provided Service to a Delivery Service.</p> <p>When this box is unchecked, changes cannot be made to the Service Item Group. For example, if the box is unchecked and the Provided Service is removed from a work order that contains the Provided Service, the Delivery Service, and the Product Item, the work order itself is cancelled. Also, when the Provided Service is added to a work order that contains the Delivery Service and the Product Item, a "VAS00002: Invalid Service Item Group Code" error is thrown.</p> <p>Note: Existing users who wish to upgrade from an older version of Yantra 7x or Sterling Supply Chain Applications to Selling and Fulfillment Foundation, Release 8.5, but wish to retain the previous functionality, should not check this box.</p>
Use Receiving Node's Calendar When Applying Receipt Processing Time	<p>This rule specifies whether the receipt processing time for future inventory and ATP Rule Processing time uses the node's receiving calendar. If no, then system days are added. If yes, then the node's receiving calendar days are added.</p>

Table 6–4 Installation Rules Window

Field	Description
Enable Extended Item Validation	<p>Check this box to extend item validations on orders to include the following:</p> <ul style="list-style-type: none"> • Effective date • Minimum and maximum quantity • Item status • Customer entitlements • IsSoldSeparately flag <p>Note: Item validation must be enabled before you extend validations.</p>
Use Deprecated Pricing Functionality	<p>Check this box to enable the deprecated pricing functionality used prior to Release 8.5 of Selling and Fulfillment Foundation.</p> <p>By default, the deprecated pricing functionality is in effect after upgrade to Release 8.5. If you uncheck this box after upgrade to enable the pricing functionality for Release 8.5, be sure to define your pricing configuration rules in the Applications Manager and convert your price lists as described in the <i>Business Center: Pricing Administration Guide</i>.</p>
Use Deprecated Data Access Policy Functionality	<p>Check this box to enable deprecated data access policies used prior to Release 8.5 across the entire application.</p> <p>To enable the deprecated data access policies for specific APIs, see the <code>yfs.properties_ext_ysc.in</code> property.</p> <p>Note: The application must be restarted to take into effect the unchecking of this configuration. Clearing of the database cache is required to take into effect the checking of this configuration. However, it is recommended to restart the application for better performance.</p>
Ship Advice Number Generation	
Use Numeric Ship Advice Number	<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	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. Note: Due to this constraint, ship advice numbers are not guaranteed to be unique.

Table 6–4 Installation Rules Window

Field	Description
Inventory Tag	
Summarize and Maintain Total Supply and Demand Values For Tag Controlled Item	Check this checkbox to summarize and maintain total supply and demand values for tag controlled items. This updates the tag descriptors and total supply and demand fields of inventory tags for tag-controlled items during inventory updates.
Display User ID	
Allow the Display User ID to be used across all enterprises	Check this checkbox if you want each User ID to be usable globally across all the enterprises in your deployment. Do not check this box if you want each User ID to be unique within each enterprise.

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 > Application Platform. The Application Platform tree displays in the side panel.
2. From the Application Platform tree, choose System Administration > Agent Criteria Groups. The Agent Criteria Group window displays in the work area.
3. Choose . The Agent Criteria Group Details pop-up window displays.

The screenshot shows a dialog box titled "Agent Criteria Group Details". It contains three text input fields stacked vertically. The first field is labeled "Agent Criteria Group", the second is labeled "Short Description", and the third is labeled "Long Description". Each field is currently empty.

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 > Application Platform. The Application Platform tree displays in the side panel.
2. From the Application Platform tree, choose System Administration > Agent Criteria Groups. The Agent Criteria Group window displays in the work area.
3. Select the Agent Criteria Group you want to modify and choose . The Agent Criteria Group Details pop-up window displays.
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 > Application Platform. The Application Platform tree displays in the side panel.
2. From the Application Platform tree, choose System Administration > Agent Criteria Groups. The Agent Criteria Group window displays 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`
- JBoss—`org.jnp.interfaces.NamingContextFactory`
- TIBCO—`com.tibco.tibcojms.naming.TibjmsInitialContextFactory`

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 displays in the work area.
2. Choose . The Initial Context Factory Details pop-up window displays.



The screenshot shows a dialog box titled "Initial Context Factory Details". It contains three text input fields:

- Initial Context Factory
- Short Description
- Long Description

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 displays in the work area.
2. Select the code you want to modify and choose . The Initial Context Factory Details pop-up window displays.
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 displays 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 Selling and Fulfillment Foundation.

To set up health monitor rules:

1. From the Application Platform tree, choose System Administration > Health Monitor Rules. The Health Monitor Rules window displays in the work area.
2. 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 *Selling and Fulfillment Foundation: System Management and Administration Guide*.
3. Choose .

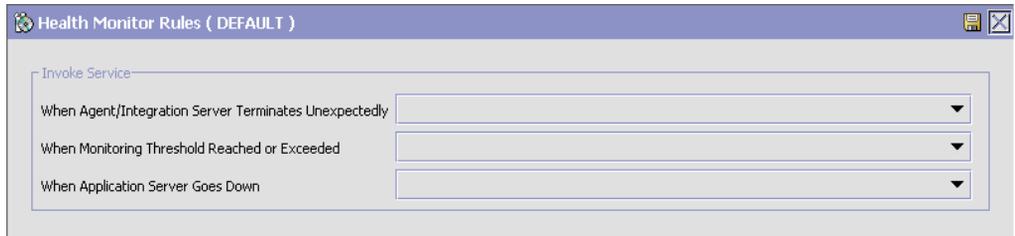


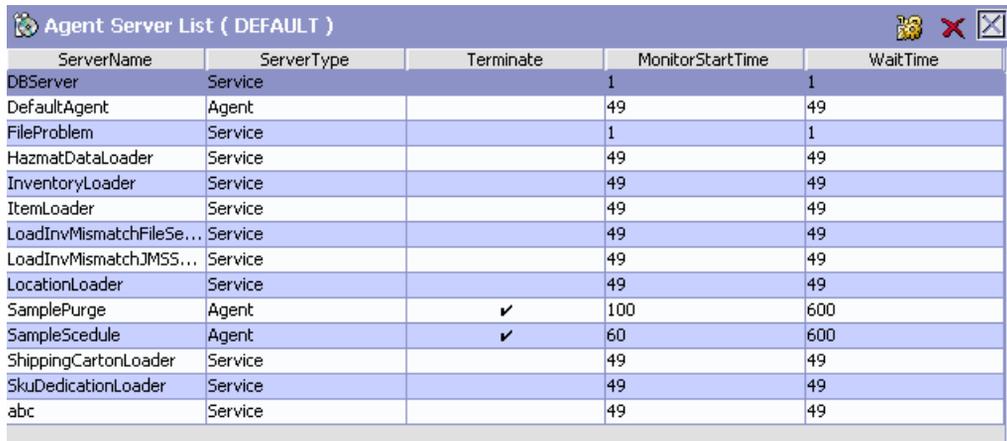
Table 6–5 Health Monitor Rules Window

Field	Description
Invoke Service	
When Agent/Integration Server Terminates Unexpectedly	Select a service to run when an agent or integration server terminates unexpectedly.
When Monitoring Threshold Reached or Exceeded	Select a service to run when a monitoring threshold for API response time, application server response time, agent pending tasks, or JMS queue number of messages is reach or exceeded for three consecutive health monitor persist intervals.
When Application Server Goes Down	Select a service to run 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 Application Platform tree in the application rules side panel, choose System Administration > Configured Servers. The Agent Server List window displays 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
SKU DedicationLoader	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.

3. The factory default agents provided in Selling and Fulfillment Foundation do 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 about the field details, see ["Adding a New Server"](#) in [Chapter 4, "Configuring Process Models"](#).

6.8 Defining Application Version

You can configure application version and code to be used to define the Qualifier Version Compatibility for your application. These application codes and versions appear in the Application Code drop-down field used when configuring qualifier version compatibility.

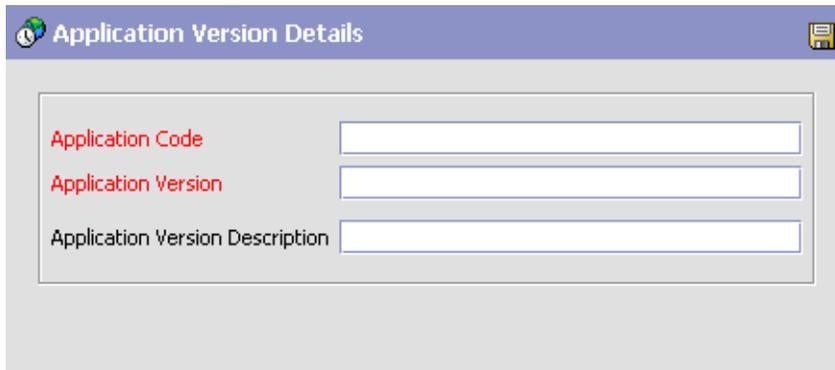
You can use the Application Version branch for:

- [Creating an Application Version](#)
- [Modifying an Application Version](#)
- [Deleting an Application Version](#)

6.8.1 Creating an Application Version

To create an application version:

1. From the tree in the application rules side panel, choose System Administration > Application Version. The Application Version window displays in the work area.
2. Choose . The Application Version Details pop-up window displays.



The screenshot shows a dialog box titled "Application Version Details". It has a title bar with a globe icon on the left and a save icon on the right. The main area contains three text input fields, each with a label to its left: "Application Code", "Application Version", and "Application Version Description".

3. In Application Code, enter the code associated with the application as you want it to appear in the drop-down menus. This is a mandatory field.
4. In Application Version, enter the application version as you want it to appear in the drop-down menus. This is a mandatory field.
5. In Application Version Description, enter a brief description of the application version.
6. Choose .

6.8.2 Modifying an Application Version

To modify an application version:

1. From the tree in the application rules side panel, choose System Administration > Application Version. The Application Version window displays in the work area.
2. Select the code you want to modify and choose . The Application Version Details pop-up window displays.
3. In Application Version Description, enter a brief description of the application version.
4. Choose .

6.8.3 Deleting an Application Version

To delete an application version:

1. From the tree in the application rules side panel, choose System Administration > Application Version. The Application Version window displays in the work area.
2. Select the application code you want to delete and choose .

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 displays in the work area.
2. Choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
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 displays 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 displays in the work area.
2. Choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
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 displays 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 displays in the work area.

2. Choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
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 displays 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 displays in the work area.
2. Choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure to modify and choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
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 displays 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 displays in the work area.

2. Choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
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 displays 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 displays in the work area.
2. Choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The Unit of Measure Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details window displays.
3. Select the applicable conversion rate and choose . The UOM Conversion Details window displays.
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 displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Conversion Details window displays.

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 displays 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 Selling and Fulfillment Foundation 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 use the Country Codes branch to set up the following common codes:

- Country codes. 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)
- Short zip code regex. This common code converts long zip codes to a simplified format. The short zip code format is used by the best

region schema matching mechanism for features such as reports, sourcing, and resource pools. T

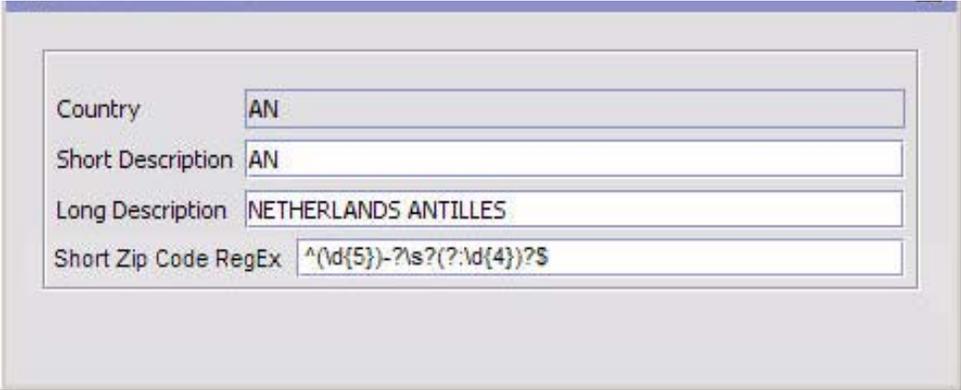
You can use the Countries branch for:

- [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 displays in the work area.
2. Choose . The Country Details pop-up window displays.



Country	AN
Short Description	AN
Long Description	NETHERLANDS ANTILLES
Short Zip Code RegEx	<code>^\d{5}-?\s?(?:\d{4})?\$</code>

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. In Short Zip Code RegEx, enter the regular expression for short zip codes for this country. If you specify a grouping for the regular expression, Selling and Fulfillment Foundation matches the zip code with the regular expression in the first parenthesis to define a short zip code. However, if no grouping is specified, Selling and Fulfillment Foundation uses the first characters that match the entire regular expression. For example, if the zip code has five characters, Selling and Fulfillment Foundation matches the first five characters to the regular expression. See the Javadocs for the `java.util.regex.Pattern` class for more information about regular expressions.

Note: If you enter a regular expression with too many characters that require escaping, you may receive a database error. Selling and Fulfillment Foundation recommends using an alternative to the POSIX character classes, which can exceed database limits due to their long length.

7. 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 displays in the work area.
2. Select the applicable country code definition and choose . The Country Details pop-up window displays.
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. In Short Zip Code RegEx, enter the regular expression for short zip codes for this country. If you specify a grouping for the regular expression, Selling and Fulfillment Foundation matches the zip code

with the regular expression in the first parenthesis to define a short zip code. However, if no grouping is specified, Selling and Fulfillment Foundation uses the first characters that match the entire regular expression. For example, if the zip code has five characters, Selling and Fulfillment Foundation matches the first five characters to the regular expression. See the Javadocs for the `java.util.regex.Pattern` class for more information about regular expressions.

Note: If you enter a regular expression with too many characters that require escaping, you may receive a database error. Selling and Fulfillment Foundation recommends using an alternative to the POSIX character classes, which can exceed database limits due to their long length.

6. 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 displays 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.

Following is the Selling and Fulfillment Foundation default language definition:

- EN (English)

You can use the Languages branch for:

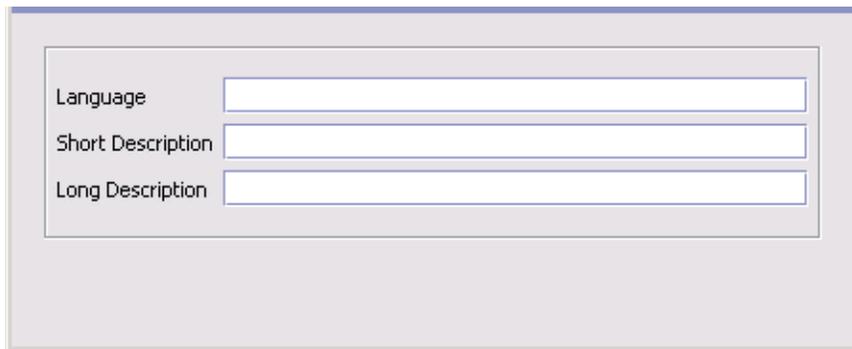
- [Creating a Language Definition](#)
- [Modifying a Language Definition](#)

- [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 displays in the work area.
2. Choose . The Language Definition Details pop-up window displays.



The screenshot shows a dialog box titled "Language Definition Details". It contains three text input fields with labels to their left: "Language", "Short Description", and "Long Description". The "Language" field is the top-most and is currently empty. The "Short Description" and "Long Description" fields are below it, also empty.

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 displays in the work area.
2. Select the applicable language definition and choose . The Language Definition Details pop-up window displays.
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 displays 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

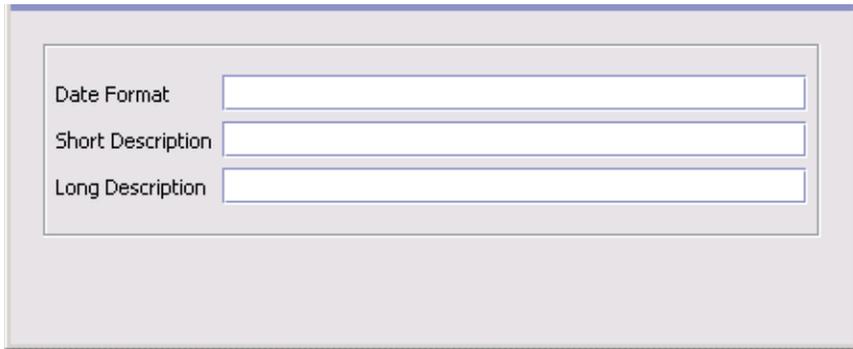
You can use the Date Formats branch for:

- [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 displays in the work area.
2. Choose . The Date Format Details pop-up window displays.



The screenshot shows a dialog box titled 'Date Format Details'. It contains three text input fields with labels to their left: 'Date Format', 'Short Description', and 'Long Description'. Each field is currently empty.

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 displays in the work area.
2. Select the applicable date format and choose . The Date Format Details pop-up window displays.
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 displays 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: Selling and Fulfillment Foundation 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:

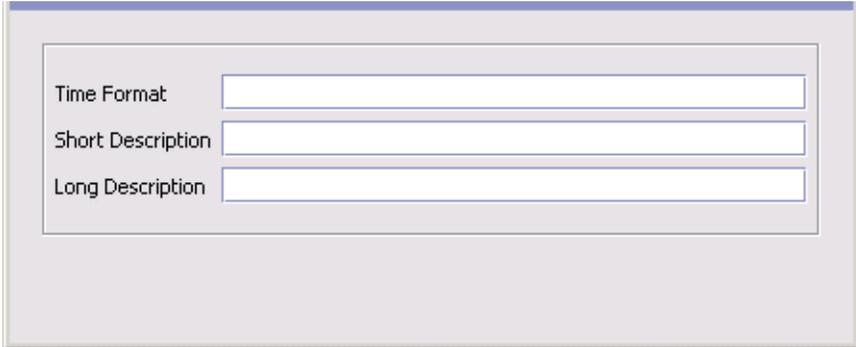
- [Creating a Time Format](#)
- [Modifying a Time Format](#)
- [Deleting a Time Format](#)

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 displays in the work area.

2. Choose . The Time Format Details pop-up window displays.



The screenshot shows a dialog box titled 'Time Format Details'. It contains three text input fields, each with a label to its left:

- Time Format
- Short Description
- 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 displays in the work area.
2. Select the applicable time format and choose . The Time Format Details pop-up window displays.
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 displays 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: Selling and Fulfillment Foundation 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 displays in the work area.
2. Choose . The Date/Time Format Details pop-up window displays.

The image 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 displays in the work area.
2. Select the applicable date/time format and choose . The Date/Time Format Details pop-up window displays.
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 displays 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

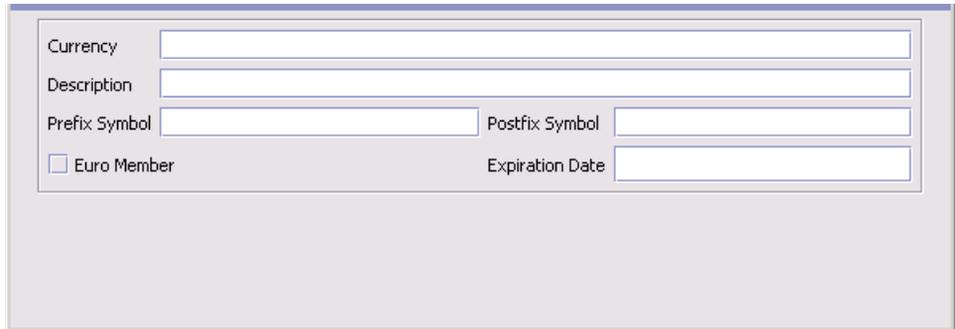
You can use the Currency Definitions branch for:

- [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 displays in the work area.
2. Choose . The Currency Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 7–17](#) for field value descriptions.
4. Choose .



The screenshot shows a dialog box titled 'Currency Details'. It contains the following fields:

- Currency:** A text input field.
- Description:** A text input field.
- Prefix Symbol:** A text input field.
- Postfix Symbol:** A text input field.
- Euro Member:** A checkbox.
- Expiration Date:** A date input field.

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).
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 displays in the work area.
2. Select the applicable currency definition and choose . The Currency Details pop-up window displays.
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 displays 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 displays 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 displays in the work area.
2. Choose . The Currency Conversion Details window displays.
3. Enter information in the applicable fields. Refer to [Table 7–18](#) for field value descriptions.
4. Choose .

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 displays in the work area.
2. Select the applicable currency conversion and choose . The Currency Conversion Details window displays.
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 displays in the work area.
2. Select the applicable currency conversion and choose .

Configuring Presentation Components

The 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 an 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 displays in the work area.
2. Choose . The Locale Details window displays.

3. Enter information in the applicable fields. Refer to [Table 9–1](#) for assistance.
4. Choose .

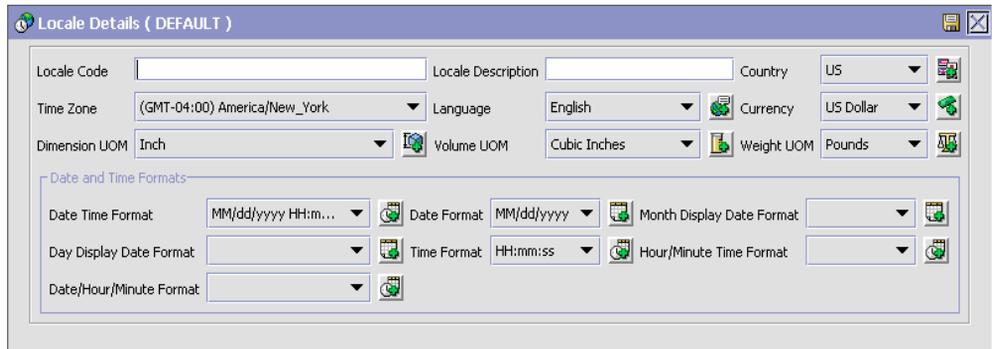


Table 9–1 *Locale Details Window*

Field	Description
Locale Code	Enter the name of the locale as you want it to appear throughout Selling and Fulfillment Foundation. Cannot contain spaces. The recommended syntax is to join the entries for Country and Language with an underscore. For example, use <i>fr_CA</i> 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 <i>CA</i> 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 <i>fr</i> 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.

Table 9–1 *Locale Details Window*

Field	Description
Weight UOM	Select the unit of measure the locale uses for weight.
Date and Time Formats	
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 displays.
Day Display Date Format	Select the date format that should be used when a day of week displays.
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 displays.
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) displays.
Client Time Format	Select the time format used in the locale. It also supports AM/PM time format. Note: This field is applicable only for Rich Client Platform applications.
Client Date Time Format	Select the date/time format used in the locale. It also supports AM/PM time format. Note: This field is applicable only for Rich Client Platform applications.

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 displays in the work area.
2. Select the applicable locale and choose . The Locale Details window displays.
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 displays 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 Selling and Fulfillment Foundation 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 Application Console.

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](#)

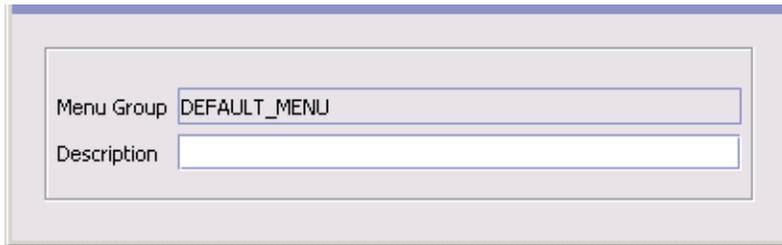
9.2.1 Saving a Menu Group as a New Menu Grouping

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 displays in the work area.

2. Select the applicable menu group and choose . The Menu Group Details pop-up window displays



The screenshot shows a dialog box titled 'Menu Group Details'. It has two text input fields. The first field is labeled 'Menu Group' and contains the text 'DEFAULT_MENU'. The second field is labeled 'Description' and is currently empty.

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 displays 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 displays 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 displays.

The image shows a configuration window with three input fields. The first field is labeled 'Menu Group' and has a dropdown arrow, currently showing 'Default Menu'. The second field is labeled 'Application' and also has a dropdown arrow, currently empty. The third field is labeled 'Description' and is a standard text input box, currently empty.

4. In Description, enter the name of the application menu as you want it to appear in the menu hierarchy interface.
5. Choose .

Note: By default, submenus are not displayed on the UI for a given menu item. You must enable submenus in the UI by adding some context parameters to the `<module_id>_config.xml` file. For more information about enabling submenus in the UI, see *Selling and Fulfillment Foundation: Customizing Console JSP Interface for End User Guide*.

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 displays 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 displays.

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 .

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 <INSTALL_DIR>/jar/<install_dir_name>/<version>/scfoundationiconsbe.jar file. Inside the .jar file, many of the icons are located in the <INSTALL_DIR>/repository/eardata/smcfs/war/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 the Resource Hierarchy tree.</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 displays 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 displays.
5. Enter information in the applicable fields. Refer to [Table 9–2the 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 displays 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 Hierarchy Tree



9.3.1 Selling and Fulfillment Foundation Resources

All Selling and Fulfillment Foundation 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 Application Console.

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 about 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>YOMD010I101</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>. • RCP Resource - <three-character pca code>RCP<sequence number>. For example, <i>YCDRCP0122</i>.
Description	<p>Literal value displayed as a label on the target screen and within the Resource Hierarchy tree. This value is stored within the resource bundle.</p>

Table 9–3 Primary Information of the Resource Details

Field	Description
URL	<p>For Selling and Fulfillment Foundation enter the uniform resource locator of the resource.</p> <p>For the Rich Client Platform applications, enter the ID of the action that the resource is associated to. This ID is the action id of the class that invokes the screen, which is defined in the <code>plugin.xml</code> file. For more information about how to create actions, see the <i>Selling and Fulfillment Foundation: Customizing the RCP Interface Guide</i>.</p>
Resource Type	<p>Sets of characteristics that define a set of features on the user interface. For the Application Consoles resources, see the following:</p> <ul style="list-style-type: none"> • Application Consoles Entity • Application Consoles Related Entity • Application Consoles Search View • Application Consoles Detail View • Application Consoles List View • Application Consoles API • Application Consoles Inner Panel • Application Consoles Action • Application Consoles Link • Application Consoles Icon <p>For the Selling and Fulfillment Foundation Rich Client Platform application resources, see the following:</p> <ul style="list-style-type: none"> • Rich Client Platform Action • Rich Client Platform Task

Table 9–3 Primary Information of the Resource Details

Field	Description
Resource Sequence	<p>The order in which screen-related literals and logic display 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 display in the UI. • Icon - Order in which icon links to other views display in the UI. • RCP Action or Task - Order in which the menu displays in a Selling and Fulfillment Foundation Rich Client Platform application.

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.

 JSP List : availability

List of JSPs used by the resource and its corresponding sub-resources

```

/im/availability/list/availability_list.jsp
/im/availability/search/availability_search_byallattr.jsp
/im/availability/detail/availability_detail_create_sales_order.jsp

```

 User List for Resource : YIMD140

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 Hierarchy tree. 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 DocumentType attribute name. This is also used to pick up literals from the resource bundle when the <code>literal</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 <INSTALL_DIR>/extensions/global/webpages directory.</p> <p>Your custom resources typically points to a file within the /extensions directory. For example, if you are changing the alert search by order search view, specify the <INSTALL_DIR>/extensions/global/webpages/em/alerts/search/alert_search_byorder.jsp file.</p> <p>As you populate the /extensions directory with custom JSP files, mimic the structure of the directory containing the standard JSP files.</p>
Output Name Space	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.
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 <INSTALL_DIR>/extensions/global/webpages directory. Your custom resources typically points to a file within the /extensions directory. For example, if you are changing the anchor page of the alert detail view, your JSP might be /extension/global/webpages/em/alerts/detail/alert_detail_anchor.jsp.</p> <p>As you populate the /extensions directory with custom JSP files, mimic the structure of the directory containing the standard JSPs.</p>
Output Name Space	N/A
Height	When this view displays in a pop-up screen, this is the height of the popup window.
Width	When this view displays 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 displays.</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.

Table 9–7 Detail View Resource Dialog Box

Control	Description
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 is 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 <code><INSTALL_DIR>/extensions/global/webpages</code> directory. Your custom resources typically point to a file within the <code>/extensions</code> directory. For example, if you are changing the alert list verbose view, specify the <code>/extensions/global/webpages/em/alerts/list/alerts_list_verbose.jsp</code> file. As you populate the <code>/extensions</code> 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

Table 9–8 List View Resource Dialog Box

Control	Description
Ignore Default API checkbox	When this option is set, any detail API that might be configured for the current entity is not invoked when the current list view displays.
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 display.
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 Selling and Fulfillment Foundation 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 of the Applications Manager.

Table 9–9 API Resource Dialog Box

Control	Description
Templates	<p>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.</p> <p>Important: note that one service should not execute the same API twice because you cannot configure multiple template elements for the same API.</p>
Invoke an API	<p>Specifies that a Selling and Fulfillment Foundation standard API is invoked in the UI for this resource. When checked, it enables the API Name drop-down list and API Template.</p>
API Name	<p>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 Service Definition Framework. If the API you select is backward compatible, the Requires Backward Compatibility checkbox is enabled.</p>
Requires Backward Compatibility	<p>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.</p>
Version	<p>Applies only to APIs running in backward compatibility mode. Provides a drop-down list of the versions for which an API is backward compatible.</p>
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>

Table 9–9 API Resource Dialog Box

Control	Description
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 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 displays. 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 Selling and Fulfillment Foundation 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>Selling and Fulfillment Foundation: Customizing Console JSP Interface for End User Guide</i>.</p>
API Type	<p>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:</p> <ul style="list-style-type: none"> • List - invokes the API from a list view • Detail - invokes the API from a detail view

Table 9–9 API Resource Dialog Box

Control	Description
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><INSTALL_DIR>/extensions/global/webpages</code> directory. Your custom resources typically point to a file within the <code>/extensions</code> directory. For example, if you are changing the person info inner panel, specify the <code>/extensions/global/webpages/pm/personinfo/detail/personinfo_detail_anchor.jsp</code> file. As you populate the <code>/extensions</code> directory with custom JSP files, mimic the structure of the directory containing the standard JSPs.
Override Entity ID	For future use.
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.

Note: Inner panels can be marked as read-only; see [Section 5.2.1.2, "Setting Inner Panels as Read-Only"](#) for more information.

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 Application Console.

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>extensions/global/webpages/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 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 displays 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 Applications Manager.

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 displays 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	The JavaScript function specified here is executed through the <code>eval()</code> function. The function body needs to be present in the <code>/extensions/global/webpages/scripts/extn.js</code> file. This JS file is included in the <code>container.jsp</code> file and is therefore available in all JSPs. Note: The <code>container.jsp</code> file is one of the 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.
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	You can specify an XML Binding here. 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.
Display Binding	You can specify an XML Binding here. 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.
Image	Relative path of the image file. For a Application Console menu item, specify the <code>com/smcfs/ycp/ui/icons/configmenu.gif</code> file.
Tooltip	You can specify an XML Binding here. When the screen displays, the XML pointed to by the binding is evaluated. If the XML attribute returns a numeric value other than zero (0), this number displays in parenthesis next to the resource description in the tooltip of this icon.

Table 9–13 Icon Resource Dialog Box

Control	Description
Alternate Image Binding	You can specify an XML Binding here. This field is 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 is used.
Alternate Image	Relative path of the alternate image file.

9.3.12 Rich Client Platform Action

In the Selling and Fulfillment Foundation Rich Client Platform, actions are defined as resources. The Rich Client Platform actions enables you to specify the ID of the action which is called when a menu item is invoked. The Rich Client Platform Action resources are permission controlled.

- **ActionID**—The identifier of the action that gets invoked when you click on a menu item.

The URL field in the resource configuration should point to the Rich Client Platform ActionID. For more information about URL field, see [Table 9–3](#).

The Rich Client Platform Action resource types can be linked to Menu items within Rich Client Platform. To configure the Rich Client Platform Action resource type with menu items, see [Section 9.2, "Defining Menus"](#).

For more information about Rich Client Platform Actions, refer to the *Selling and Fulfillment Foundation: Customizing the RCP Interface Guide*.

9.3.13 Rich Client Platform Task

Each Rich Client Platform Task resource type is associated with a resource identifier. The Rich Client Platform Task resource types are permission controllable. The user can view only those Rich Client Platform tasks for which the user has permissions. The Rich Client Platform Task resources types are the logical representation of related tasks within the Rich Client Platform. The Rich Client Platform task contains a set of actions. When you create the related tasks, you must specify the identifier of the resource that you want to

associate with the related task. Based on the permissions assigned for the resource, the related tasks view displays to the user.

The URL field in the resource configuration should point to the Rich Client Platform ActionID. For the URL field description, see [Table 9–3](#).

For more information about Related Tasks and Rich Client Platform Actions, see the *Selling and Fulfillment Foundation: Customizing the RCP Interface Guide*.

9.4 Defining Themes

You can define the labels that identify themes used in Selling and Fulfillment Foundation. All themes used by Selling and Fulfillment Foundation are defined centrally through CSS and XML files. Themes contain the standard Selling and Fulfillment Foundation 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 displays in the work area.
2. Choose . The Theme Details pop-up window displays.

The image shows a dialog box titled 'Theme Details' with three text input fields. The first field is labeled 'Theme', the second 'Short Description', and the third 'Long Description'. Each field is currently empty.

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 *Selling and Fulfillment Foundation: 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 displays in the work area.
2. Select the applicable theme and choose . The Theme Details pop-up window displays.
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 displays 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 Selling and Fulfillment Foundation 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.

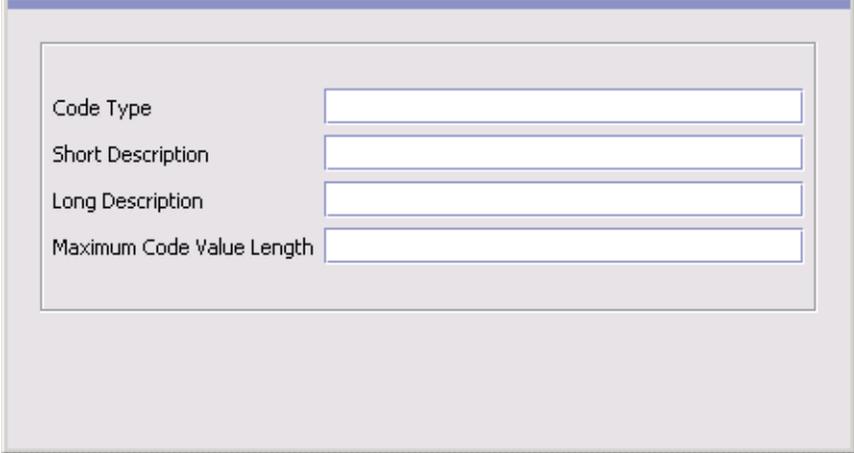
You can use the Custom Common Code Types branch for:

- [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 displays in the work area.
2. Choose . The Code Type Details pop-up window displays.



The screenshot shows a form with four input fields, each with a label to its left:

- Code Type
- Short Description
- Long Description
- Maximum Code Value Length

3. 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 does not conflict with the Selling and Fulfillment Foundation-defined common codes. This value must be passed to the `getCommonCodeList()` API.

4. In Short Description, enter a brief description of the theme.
5. In Long Description, enter a more detailed description of the theme.
6. 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.
7. 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 displays in the work area.
2. Select the applicable custom common code type and choose . The Code Type Details pop-up window displays.
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 displays 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:

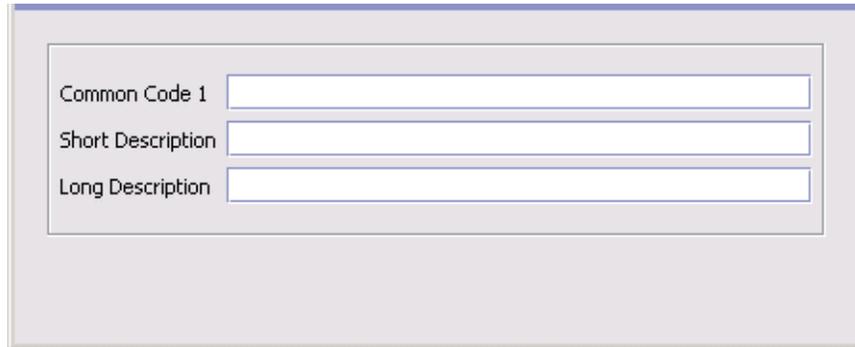
- [Adding Values to a Custom Common Code](#)
- [Modifying a Custom Common Code's Values](#)
- [Deleting a Custom Common Code's Values](#)

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 displays in the work area.

2. Select the applicable custom common code and choose . The Common Code Values pop-up window displays.
3. Choose . The (*Common Code*) Details pop-up window displays.



The screenshot shows a dialog box with a light gray background. Inside, there is a white rectangular area containing three text input fields. The first field is labeled 'Common Code 1', the second is labeled 'Short Description', and the third is labeled 'Long Description'. Each field has a thin blue border and is currently empty.

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 displays in the work area.
2. Select the applicable custom common code and choose . The Common Codes pop-up window displays.
3. Locate the applicable code value and choose . The common code's Details pop-up window displays.
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 displays in the work area.
2. Select the applicable custom common code and choose . The Common Codes pop-up window displays.
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 Selling and Fulfillment Foundation. 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 displays 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.

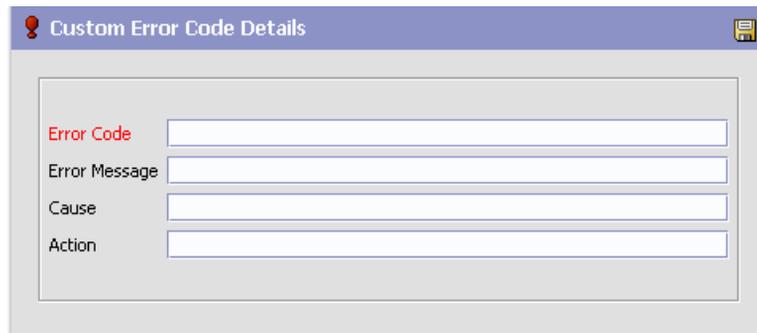
You can use the Custom Error Codes list for:

- [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 displays 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 displays 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 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 displays in the work area.
2. Choose  and a list of custom error codes display.
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 Selling and Fulfillment Foundation 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 displays in the work area.
2. Choose . The Protocol Details pop-up window displays.



The image shows a pop-up window titled 'Protocol Details'. It contains two text input fields. The first field is labeled 'Protocol Name' and the second field is labeled 'Description'. Both fields are currently empty.

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 displays in the work area.
2. Select the applicable protocol and choose . The Protocol Details pop-up window displays.
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 displays 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 displays in the work area.
2. Choose . The Document Format Details pop-up window displays.



The image shows a dialog box with a light gray background and a thin border. Inside the dialog, there are 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 displays in the work area.
2. Select the applicable document format and choose . The Document Format Details pop-up window displays.
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 displays 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 displays in the work area.
2. Choose  from the top portion of the screen. The Business Document Details pop-up window displays.



Document ID	<input type="text"/>
Document Name	<input type="text"/>
Description	<input type="text"/>

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 displays in the work area.
2. Select the applicable business document and choose . The Business Document Details pop-up window displays.
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:

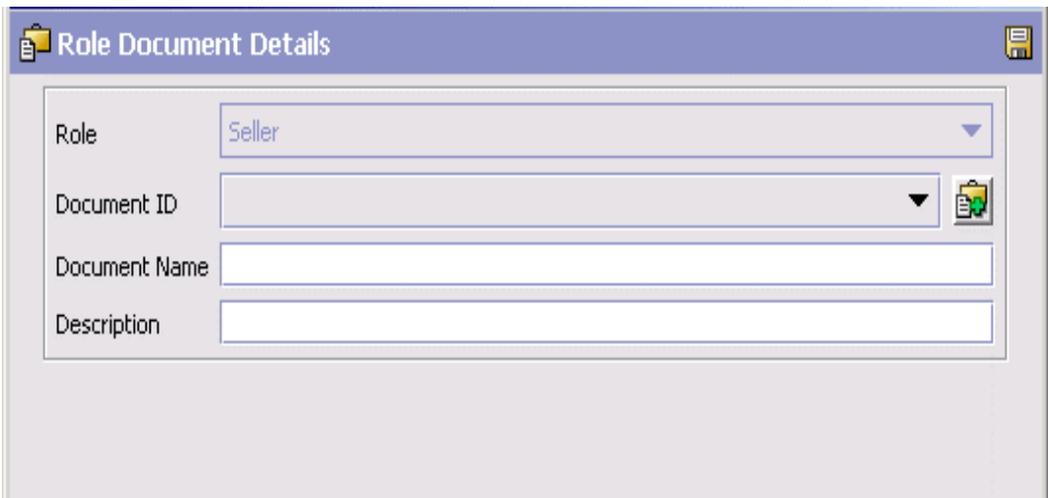
1. From the tree in the application rules side panel, choose Communication > Business Document. The Business Document List window displays in the work area.
2. Select the applicable document format and choose .

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 displays 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 displays.



Role Document Details

Role:

Document ID: 

Document Name:

Description:

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 Selling and Fulfillment Foundation, they may identify entity values differently than Selling and Fulfillment Foundation and each other. For example, a carrier code may be called `UPS` in Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation 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:

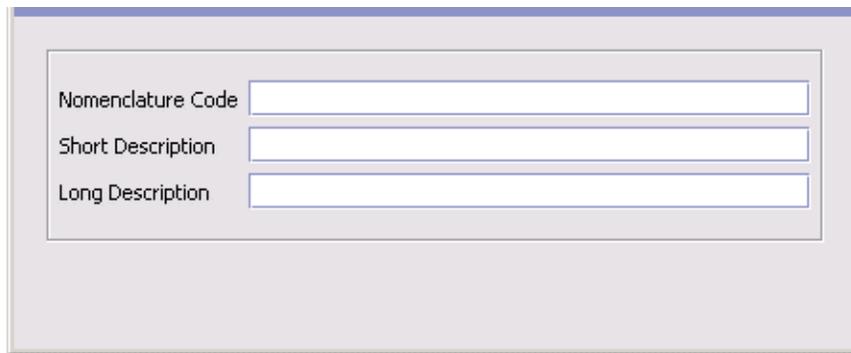
- [Defining Nomenclature Codes](#)
- [Defining Nomenclature Definitions](#)
- [Defining Nomenclature Configuration](#)

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 > Application Platform. The Application Platform tree displays in the side panel.
2. From the Application Platform tree, choose Nomenclature > Nomenclature Codes. The Nomenclature Codes window displays in the work area.
3. Choose . The Nomenclature Code Details pop-up window displays.



Nomenclature Code

Short Description

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 displays in the work area.
2. Choose . The Nomenclature definition details pop-up window displays. Enter information, using the [Table 11–1, "Nomenclature Definition Menu"](#) for help.
3. Choose .

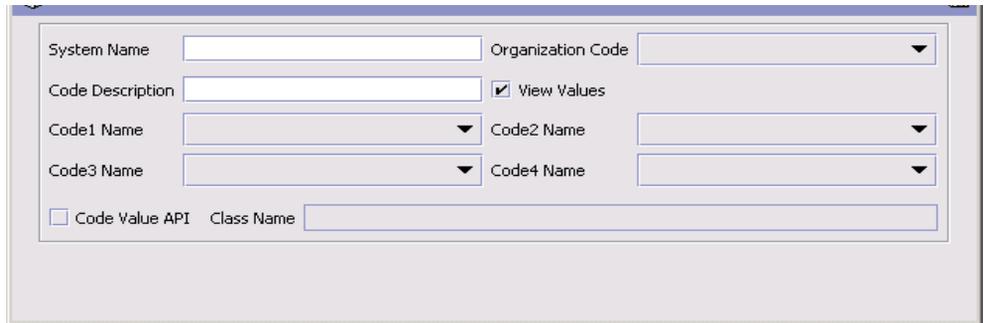


Table 11–1 Nomenclature Definition Menu

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 STERLING 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.

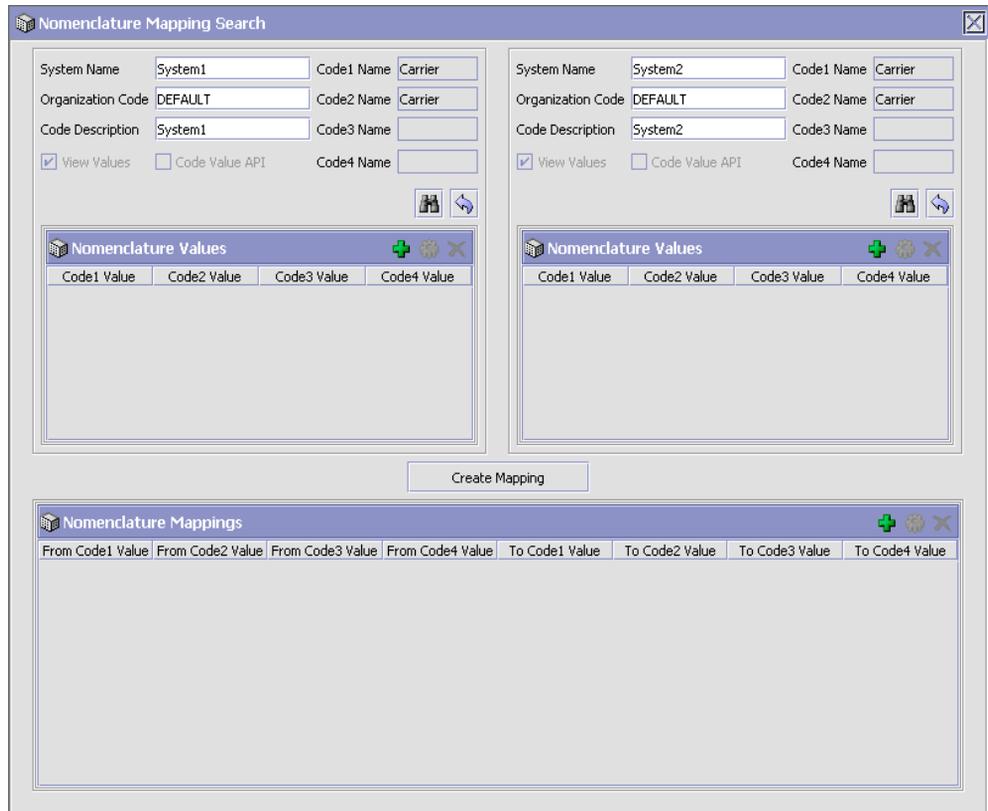
Table 11–1 Nomenclature Definition Menu

Property	Description
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.
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 STERLING (the Selling and Fulfillment Foundation-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 displays in the work area.
2. Select the two Entities that you want to map and choose the Mapping button. The Nomenclature Mapping panel displays.



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 .

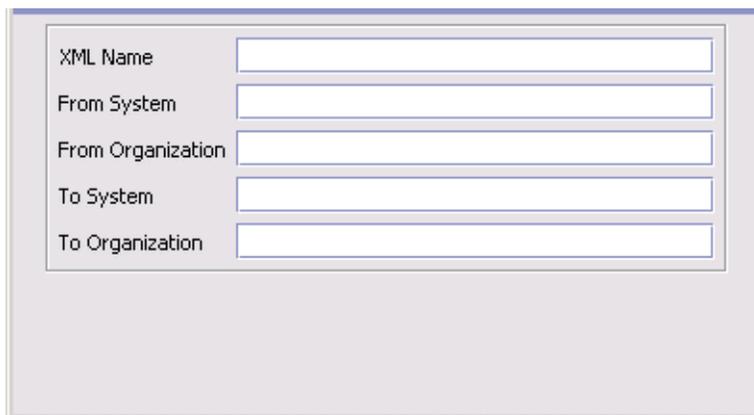
5. To create a mapping between the two entities, select a row from the list of values displayed in the from/to system and choose the Create Mapping button.

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 Selling and Fulfillment Foundation 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 displays in the work area.
2. Choose . The Create Nomenclature Configuration pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 11–2, "Nomenclature Configuration"](#) for field level descriptions.

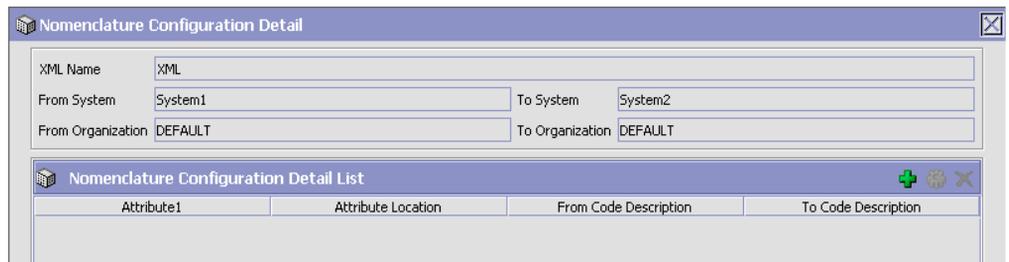


XML Name	<input type="text"/>
From System	<input type="text"/>
From Organization	<input type="text"/>
To System	<input type="text"/>
To Organization	<input type="text"/>

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 displays.



5. Choose . The Configuration Detail pop-up window displays 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.

XML Name: XML

Attribute Location: []

Attribute1: [] New Attribute1: []

Attribute2: [] New Attribute2: []

Attribute3: [] New Attribute3: []

Attribute4: [] New Attribute4: []

Mapping Type: Nomenclature Constant Abort Processing On Error

Nomenclature

From Code Description	To Code Description	Default Values
[]	[]	<input type="checkbox"/> Use Default Values
[]	[]	[]
[]	[]	[]
[]	[]	[]
[]	[]	[]

Constant

Constant1: [] Constant2: []

Constant3: [] Constant4: []

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

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 displays in the work area.

2. Choose . The Queue Details window displays.
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.
Queue Group	Enter the group this queue is a part of.

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](#)

Note: The Exception Monitor has to be running in the background for the above configuration to work.

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.
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.

Note: You cannot resolve unassigned alerts.

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.



The screenshot shows the configuration interface for the 'Unresolved Alerts' tab. It features four input fields:

- Escalate Alert After:** A text input field followed by the label 'Hours'.
- Move Alert To Different Queue:** A dropdown menu with a small icon of a queue.
- Raise Action:** A dropdown menu with a small icon of a plus sign.
- Re-Alert After:** A text input field followed by the label 'Hours'.

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 5.1, "Defining Users"](#).

To view and modify users subscribed to a queue:

1. In the Queue Details window, choose the User List tab. The User Detail list displays.
2. Select the applicable user and choose . The User Detail window displays. For more information about how to modify a user, see [Section 5.1.1, "Creating and Modifying a User"](#).
3. To remove a user from the queue choose .



User ID	User Name	Organization
Barb	Barbara Ann	DEFAULT
Sterling	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 displays in the work area.

2. Enter applicable search criteria and choose . The Queue list displays.
3. Select the applicable queue and choose . The Queue Details window displays.
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 displays in the work area.
2. Enter applicable search criteria and choose . The Queue list displays.
3. Select the applicable queue and choose .

Configuring Region Definitions

Region definitions allows you to configure the components that are used by the Selling and Fulfillment Foundation geography engines. The individual components consisting of regions and region levels can be used to create region schemas that can then be used throughout Selling and Fulfillment Foundation 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 displays 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 .

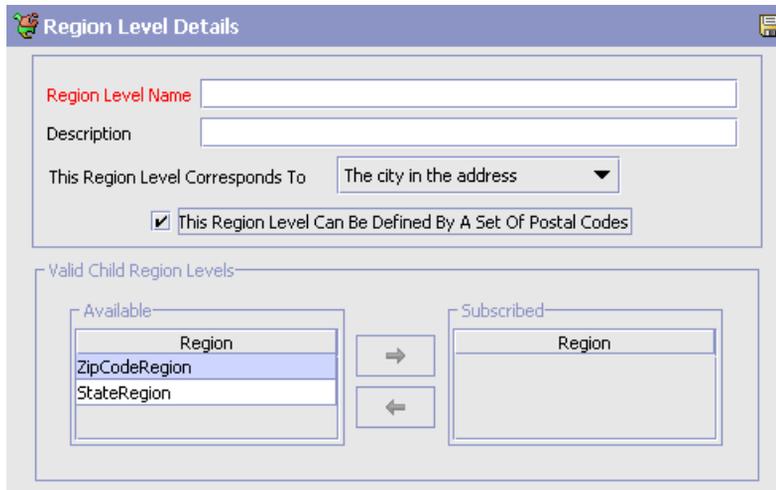


Table 13–1 *Region Level Details Window*

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 about region matching, see Section 13.2, "Defining Region Match Preferences" .

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 displays 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 displays 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 about region matching, see the *Selling and Fulfillment Foundation: Product Concepts Guide*.

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 displays in the work area.
2. Enter information in the applicable fields. Refer to [Table 13–2](#) for field value descriptions.
3. Choose .

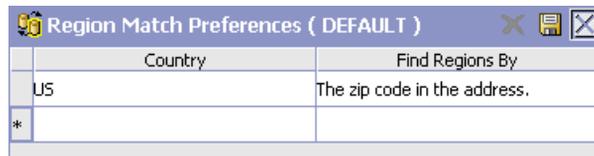


Table 13–2 Region Match Preferences

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 displays 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 displays 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 displays.
2. Enter information in the applicable fields. Refer to [Table 13–4](#) for field value descriptions.
3. Choose .

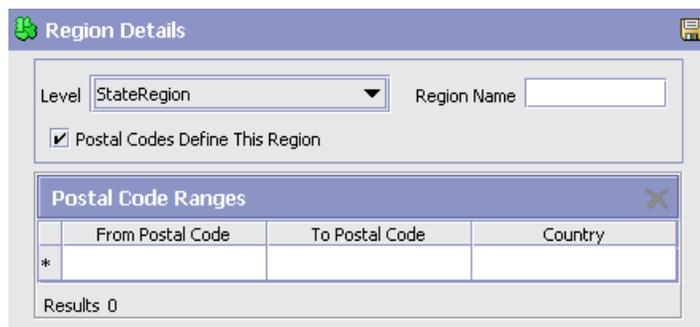


Table 13–4 *Region Details Window*

Field	Description
Level	Select the region's level. For more information about region levels, see Section 13.1, "Defining Region Levels" .
Region Name	Enter the name of the region. If a region level has been selected that maps to the 'Country' field, this field provides 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".</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 displays.
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 displays 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 delete a region schema:

1. From the tree in the application rules side panel, choose Region Definition > Region Schemas. The Region Schemas window displays 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 Selling and Fulfillment Foundation. An individual unit is defined as a sub type for a device type.

For example, device types include RF scanners, printers, and weighing scale.

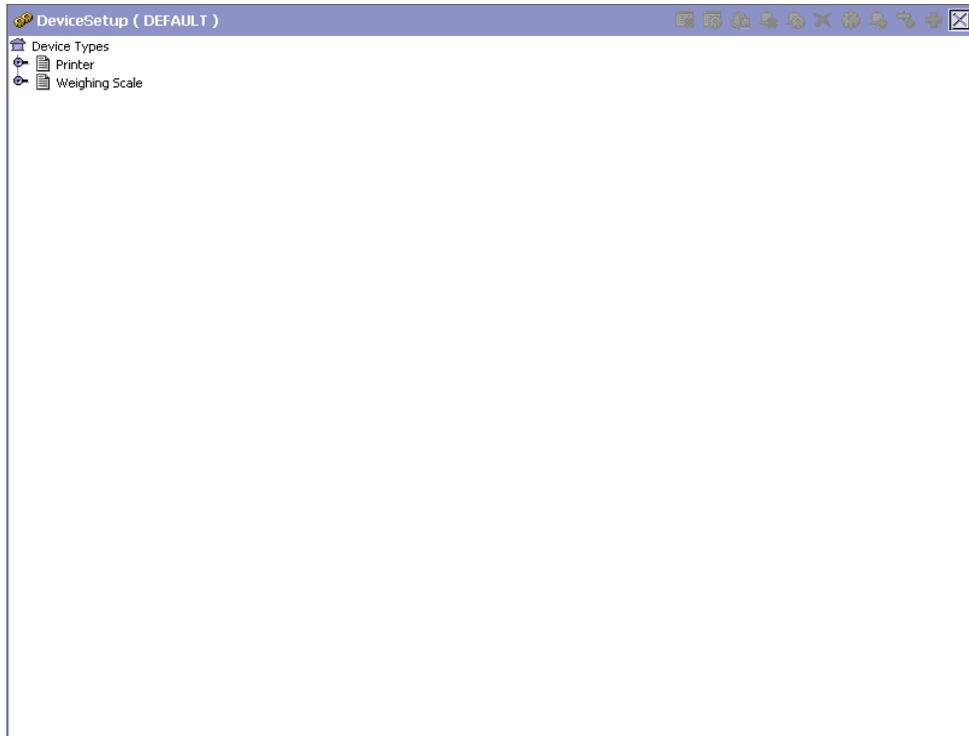
Use Device Type for:

- [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 displays.



2. In the Device Setup window, choose . The Device Type pop-up window displays.
3. Enter information in the applicable fields. Refer [Table 14–1](#) for field value descriptions.
4. Choose .

The image shows a pop-up window with a light gray background. At the top left, there are two text input fields. The first field is labeled 'Device Type' and the second is labeled 'Description'. Both fields are empty and have a standard text box appearance with a thin border and a light blue highlight when active.

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 displays with the list of Device Types.
3. Select the Device Type to be modified. Choose .
4. The Device Type pop-up window displays.
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 displays 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, Eltron, Unimark, 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 displays.
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 displays.
4. Enter information in the applicable fields. Refer [Table 14–2](#) for field value descriptions.
5. Choose .

Device Sub Type

Device Type:

Device Sub Type:

Description:

Printer Type Document Association

Printer Type Document Association		
	Print Document	Print Document Description
*		

Results 0 of 0

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.
Printer Type Document Association	
Note: This panel is available when the value of the Device Type is set to "Printer."	
Printer Type Document Association	Select which print documents you would like to associate with the selected printer.

Note: If you are configuring a new Device Sub Type for printing the FedEx Carrier Label, ensure that you map the value of the new Device Sub Type in the YCS Mapping table. For more information about the YCS Mapping table, see the *Parcel Carrier: Adapter Guide*.

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 displays.
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 displays.
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 displays.
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 has all the nine printers configured as devices.

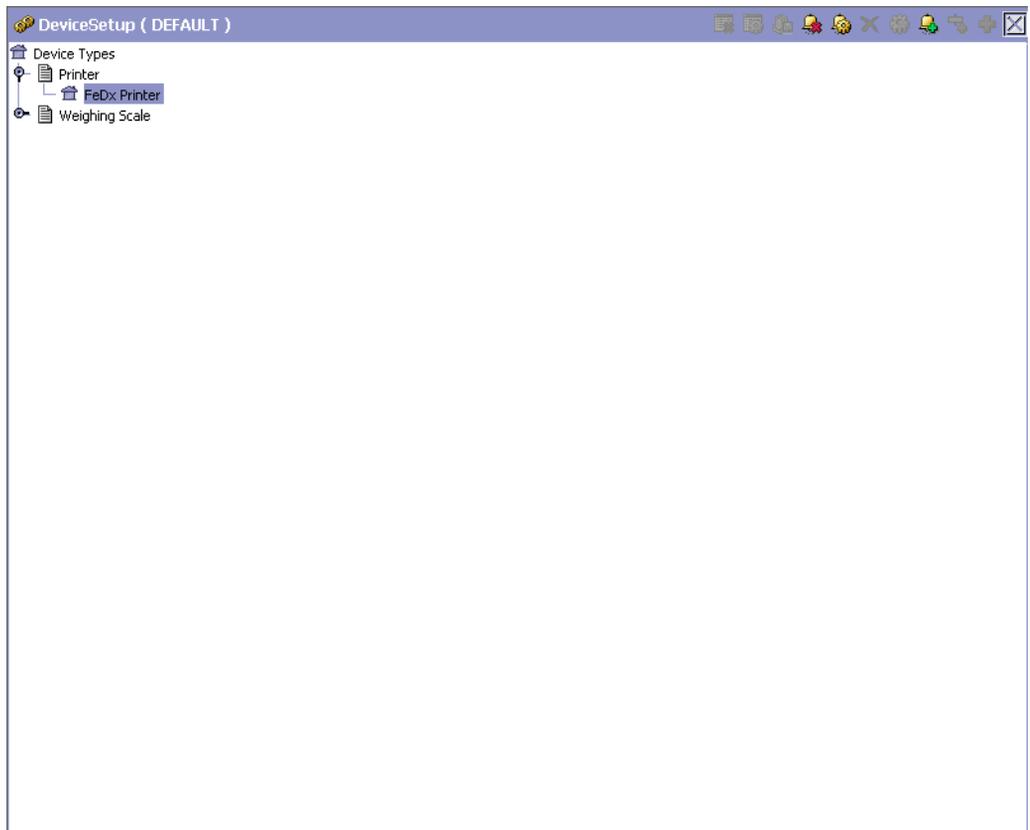
Use Device for:

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

The screenshot shows a configuration window with the following fields and table:

- Device Type: Printer
- Device Sub Type: Laserjet
- Device ID: (empty text box)
- Additional Attributes section containing a table:

Parameter Name	Parameter Value
PrinterAlias	HP LaserJet 4300 PCL 6
PrintServerHostName	dev3pty
PrintServerPort	2723

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" .

Selling and Fulfillment Foundation 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 the Selling and Fulfillment Foundation 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 about including this variable, see the *Selling and Fulfillment Foundation: Customization Guide*.

Note: The DropDirectory attribute appears in the Applications Manager only if the `yfs.software.tcpip.sockets` property is set to `N` in the `<INSTALL_DIR>/properties/customer_overrides.properties` file.

For information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

- **PrinterAlias** - The printer alias as configured in the Software printer setup.

Note: While setting up a Printer device in Selling and Fulfillment Foundation, 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, ensure that the Printer Alias on Selling and Fulfillment Foundation 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 appear in the Applications Manager only if the `yfs.software.tcpip.sockets` property is set to `Y` in the `<INSTALL_DIR>/properties/customer_overrides.properties` file.

For information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

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, see 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. An HP LaserJet printer has 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 displays.
3. Enter information in the applicable fields. Refer [Table 14–4](#) for field value descriptions.
4. Choose .

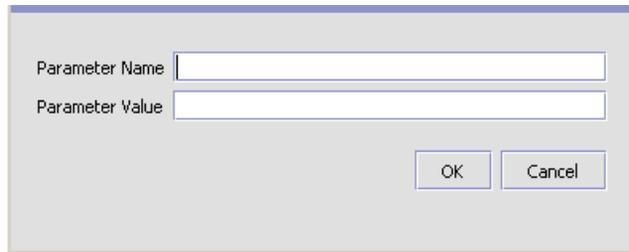


Table 14–4 *Criteria Parameter Details Pop-up Window*

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 displays.
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 displays. Select the Device to be copied to create a new device.
4. Choose . The Device pop-up window displays.
5. Enter information in the applicable fields. Refer to [Table 14–2](#) for field value descriptions.
6. Choose .

Selling and Fulfillment Foundation 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 the Selling and Fulfillment Foundation 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 about including this variable, see the *Selling and Fulfillment Foundation: Customization Guide*.

Note: The DropDirectory attribute appears in the Applications Manager only if the `yfs.ftware.tcpip.sockets` property is set to `N` in the `<INSTALL_DIR>/properties/customer_overrides.properties` file.

For information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

- PrinterAlias - The printer alias as configured in the Loftware printer setup.

Note: While setting up a Printer device in Selling and Fulfillment Foundation, 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, ensure that the Printer Alias on Selling and Fulfillment Foundation 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 appear in the Applications Manager only if the `yfs.software.tcpip.sockets` property is set to `Y` in the `<INSTALL_DIR>/properties/customer_overrides.properties` file.

For information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

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"](#).

Note: The ClassName for the Mettler Toledo Weighing Scale is `com.yantra.ycp.ui.io.YCPToledoPSImpl`. For specifications pertaining to the other attributes, see 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 displays.
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 displays. Select the Device to be modified.
4. Choose . The Device pop-up window displays.
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 displays.
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 displays. 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.

Selling and Fulfillment Foundation 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

Selling and Fulfillment Foundation 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"](#).

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. Selling and Fulfillment Foundation provides a 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:

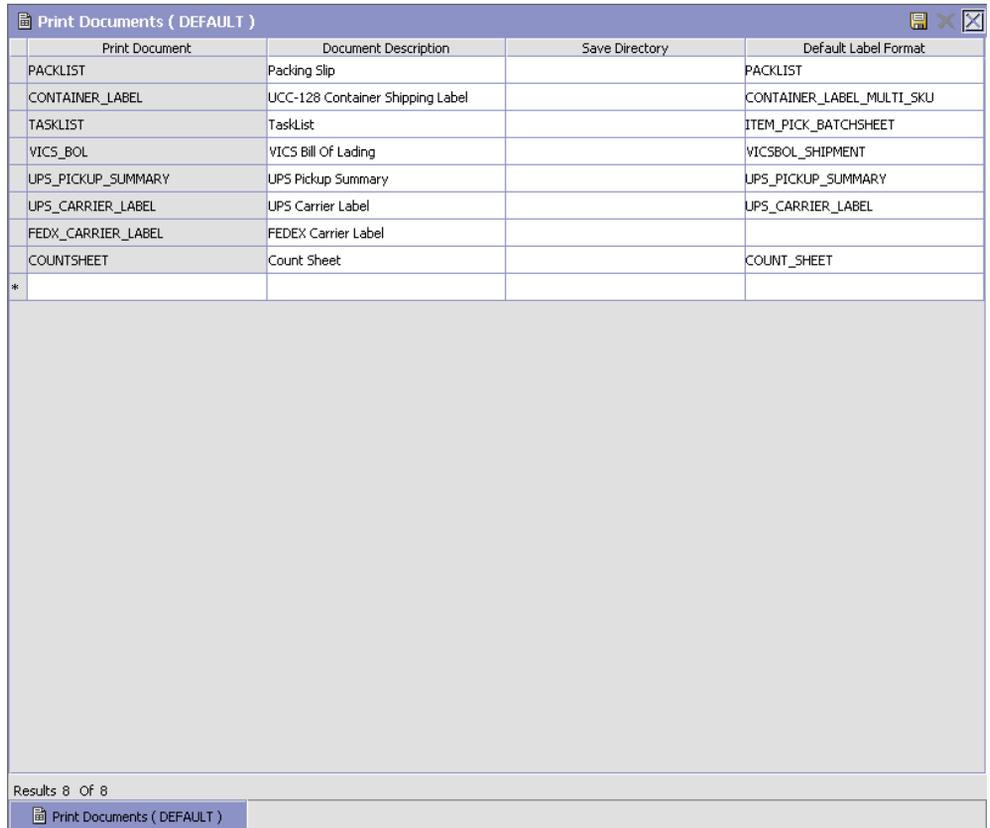
- [Creating a Print Document](#)
- [Modifying a Print Document](#)
- [Deleting a Print Document](#)

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 displays with Selling and Fulfillment Foundation 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 set in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties 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 Software 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 displays 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 Selling and Fulfillment Foundation-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 displays with a list of print documents.
2. Choose the Print Document to be deleted.
3. Choose .

It is recommended that the Selling and Fulfillment Foundation-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 the Selling and Fulfillment Foundation-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.

See the *Selling and Fulfillment Foundation: Installation Guide* for further information about installing and configuring Loftware Label Manager™.

Selling and Fulfillment Foundation 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). Selling and Fulfillment Foundation provides data flow for the standard documents provided.

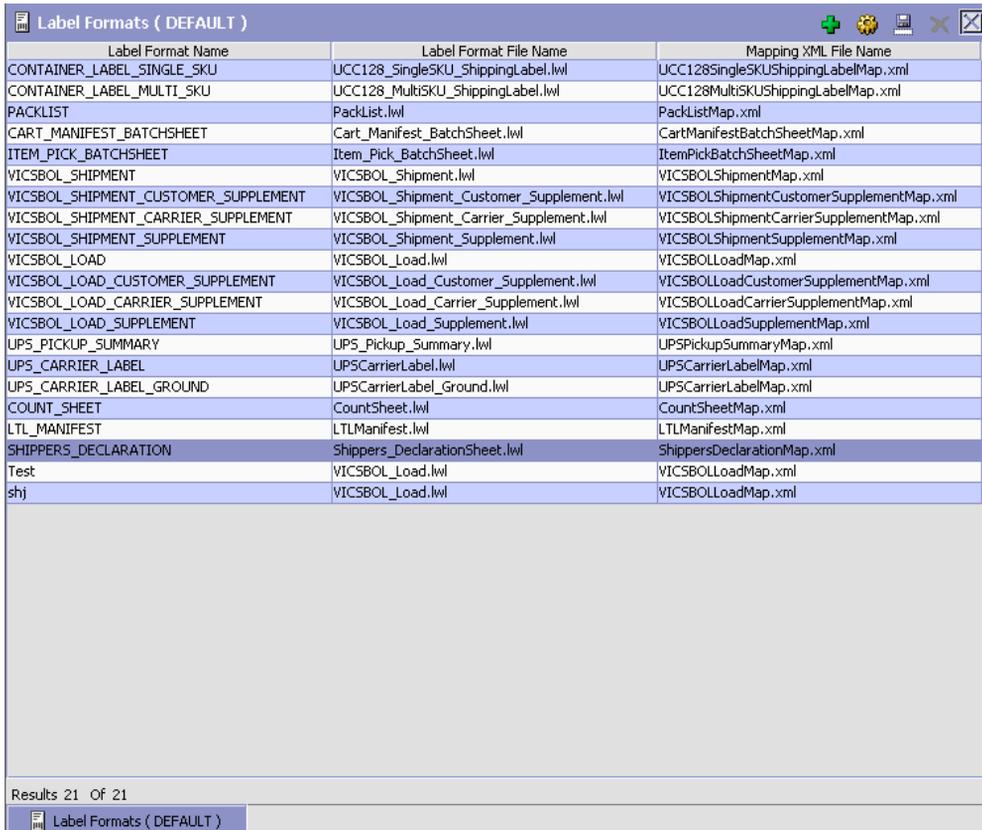
Use Label Format set-up for:

- [Creating a Label Format](#)
- [Modifying a Label Format](#)
- [Deleting a Label Format](#)

15.2.1 Creating a Label Format

To create a label format:

1. From the tree in the application rules side panel, choose Prints > Label Formats. The Label Formats window displays with the Selling and Fulfillment Foundation default label formats.



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

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)

2. In the Label Formats window, choose . The Label Details pop-up window displays.
3. Enter information in the applicable fields. Refer [Table 15–2](#) for field value descriptions.

4. Choose .

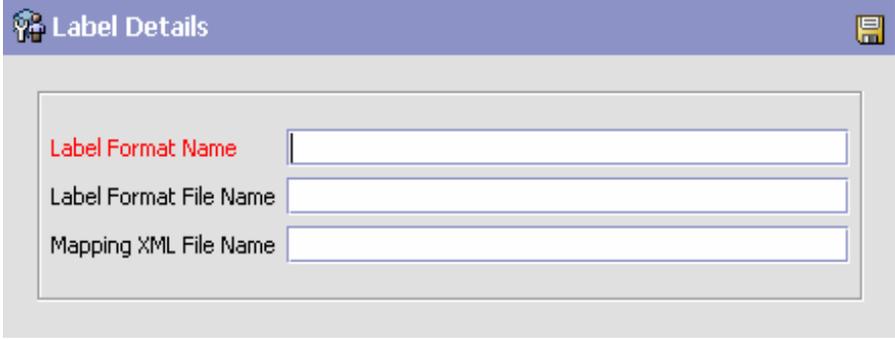


Table 15–2 Label Details Pop-up Window

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 Software Label Manager™.
Label Format File Name	Enter the name of the Software™ 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 displays 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 the Selling and Fulfillment Foundation-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 displays with a list of label formats.
2. Choose the Label Format to be deleted.
3. Choose .

It is recommended that the Selling and Fulfillment Foundation-provided standard label formats are not deleted.

16

Managing Password Policies

Selling and Fulfillment Foundation provides an in-built and flexible password management policy for controlling password use and behavior. A password policy is a set of rules to define, control, and manage user passwords. A set of default rules is provided; however, you can configure your own rules for the password policy.

For detailed information about managing password policies, see the *Selling and Fulfillment Foundation: Password Policy Management Guide*.

Configuring 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.

When configuring alerts, you can create exceptions to classify them in one or more types. When creating exception types, you can assign priority and high priority number to the exception types. You can also specify whether or not the consolidation for an exception type is required. You can configure consolidation for an exception type by specifying the consolidation window and consolidation User Exit (UE) implementation. You can configure the context sensitive resolution screen for an exception type by specifying the resolution form and resolution UE implementation. In addition, you can define some additional configurations for an exception type by specifying the configuration form and list form.

You have to associate an exception type to an application such as PCA or a role such as BUYER and SELLER. You can configure one or more exception types for an organization. You can also create routing rules for an organization level exception type and use a particular routing logic for routing these exception types to a specific alert queue.

You can use the Alerts branch for:

- [Defining Exception Types](#)
- [Defining Exception Type Role](#)
- [Defining Organization Exception Type Configuration](#)

17.1 Defining Exception Types

You can specify exception type details here. You can create new exception types and classify them in one or more types.

Use Alerts set-up for:

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

17.1.1 Creating an Exception Type

To create an exception type:

1. From the tree in the application rules side panel, choose Alerts > Exception Type. The Alert Type window displays in the work area.
2. Choose . The Exception Type Details window displays.
3. Enter information in the applicable fields. Refer to [Table 17–1](#) for field value descriptions.
4. Choose .

Figure 17–1 Exception Type Details Window

Table 17–1 Exception Type Details Pop-Up Window

Field	Description
Exception Type	Enter name for the new exception type to be created. This is a mandatory field.
Exception Type Description	Enter a brief description of the exception type. This is a mandatory field.
Priority	Enter the priority for the exception type.
High Priority Number	Enter the threshold number for the high priority exceptions of this type.
Expiration Days	Enter the number of days after which the exception type should expire.
Followup Hours	Enter the number of hours in which the exception type should be followed up.
Queue	Select the name of the alert queue to which the exception type is routed. For more information about creating alert queues, see Chapter 12, "Configuring Alert Queues" .
Owner Role	Select the role to which the organization belongs from the drop-down list.
Configurable	Check this box if you want to define additional configurations for the exception type.

Table 17–1 Exception Type Details Pop-Up Window

Field	Description
Consolidation Hook	Enter the name of the class which provides the implementation for the consolidation of the exception type.
Resolution Hook	Enter the name of the class which provides the implementation for the resolution of the exception type.
Consolidation Required	Check this box if you want to consolidate the exceptions of this type.
Consolidation Window	Enter the information about the window that is to be used for consolidating exceptions of this type. The valid values are: HOUR, DAY, and FOREVER.
Configuration Form	Enter the information about the form that is to be used to capture the additional configuration for the exception type.
Resolution Form	Enter the information about the form to be launched to resolve this type of exception.
List Form	Enter the information about the form to list the exceptions for the exception type.

17.1.2 Modifying an Exception Type

Once an exception type has been created, it can be modified.

To modify an exception type:

1. From the tree in the application rules side panel, choose Alert > Exception Type.
2. The Alert Type window displays with the list of Alert Types.
3. Select the Alert Type to be modified. Choose .
4. Enter information in the applicable fields. Refer [Table 17–1](#) for field value descriptions.
5. Choose .

17.1.3 Deleting an Exception Type

To delete an exception type:

1. From the tree in the application rules side panel, choose Alert > Exception Type.
2. The Alert Type window displays with the list of Alert Types.
3. Select the Alert Type to be deleted. Choose .

17.2 Defining Exception Type Role

You can specify a list of exception types for an application, such as PCA, and a role, such as BUYER or SELLER, that the application supports.

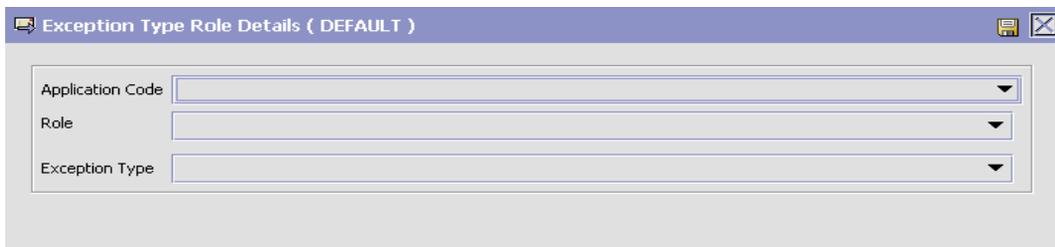
Use Alerts set-up for:

- [Creating an Exception Type Role](#)
- [Modifying an Exception Type Role](#)
- [Deleting an Exception Type Role](#)

17.2.1 Creating an Exception Type Role

To create an exception type role:

1. From the tree in the application rules side panel, choose Alerts > Exception Type Role. The Exception Type Role window displays in the work area.
2. Choose . The Exception Type Role Details window displays.
3. Enter information in the applicable fields. Refer to [Table 17–1](#) for field value descriptions.
4. Choose .



Exception Type Role Details (DEFAULT)

Application Code

Role

Exception Type

Table 17–2 Exception Type Role Details Pop-Up Window

Field	Description
Application Code	Select the code of the application. This is a mandatory field.
Role	Select the role. For example: BUYER, NODE, and so forth.
Exception Type	Select the exception type that you want to associate with an application and role.

17.2.2 Modifying an Exception Type Role

Once an exception type role has been created, it can be modified.

To modify an exception type role:

1. From the tree in the application rules side panel, choose Alert > Exception Type Role.
2. The Exception Type Role window displays with the list of Exception Type roles.
3. Select the Exception Type role to be modified. Choose .
4. Enter information in the applicable fields. Refer [Table 17–2](#) for field value descriptions.
5. Choose .

17.2.3 Deleting an Exception Type Role

To delete an exception type role:

1. From the tree in the application rules side panel, choose Alert > Exception Type Role.
2. The Exception Type Role window displays with the list of Exception Type roles.
3. Select the Exception Type role to be deleted. Choose .

17.3 Defining Organization Exception Type Configuration

Organization Exception Type Configuration allows you to activate or override additional parameters for an exception type at an organization level. These attributes are used by the alert routing service builder component. You can also create multiple routing rules for an organization level exception type. The routing rules indicate the type of routing. When creating a routing rule, you can specify the routing logic that you want to use for routing the organization level exception types to a particular queue.

Use Alerts set-up for:

- [Creating an Exception Type for an Organization](#)
- [Modifying an Exception Type for an Organization](#)
- [Creating an Exception Routing Rule](#)
- [Modifying an Exception Routing Rule](#)
- [Deleting an Exception Routing Rule](#)
- [Deleting an Exception Type for an Organization](#)

17.3.1 Creating an Exception Type for an Organization

To create an exception type for an organization:

1. From the tree in the application rules side panel, choose Alerts > Organization Exception Type Configuration. The Exception Type for Organization window displays in the work area.
2. Choose . The Exception Type for Organization Details window displays.
3. Enter information in the applicable fields. Refer to [Table 17–3](#) for field value descriptions.
4. Choose .

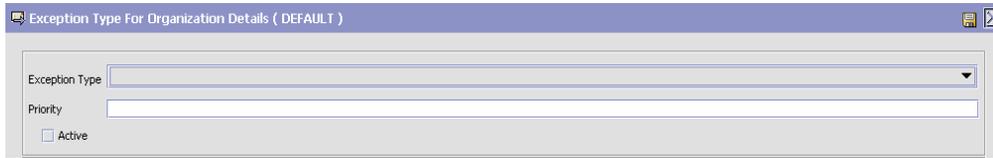


Table 17–3 *Exception Type for Organization Details Pop-Up Window*

Field	Description
Exception Type	Select the exception type that you want to associate with an organization.
Priority	Enter the priority for exceptions of this type.
Active	Check this box if you want to activate exception of this type for an organization.

17.3.2 Modifying an Exception Type for an Organization

Once an exception type for an organization has been created, it can be modified.

To modify exception type for an organization:

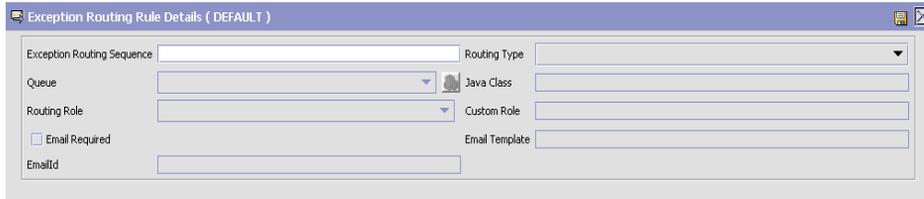
1. From the tree in the application rules side panel, choose Alert > Organization Exception Type Configuration.
2. The Organization Exception Type Configuration window displays with the list of Exception Type configured for an organization.
3. Select the Exception Type to be modified. Choose .
4. Enter information in the applicable fields. Refer [Table 17–3](#) for field value descriptions.
5. Choose .

17.3.3 Creating an Exception Routing Rule

To create an exception routing rule for an organization level exception type:

1. From the tree in the application rules side panel, choose Alert > Organization Exception Type Configuration. The Organization Exception Type for Organization window displays in the work area.

2. Select the applicable exception type and choose . The Exception Type for Organization Details window displays.
3. Choose . The Exception Routing Rule Details window displays.
4. Enter information in the applicable fields. Refer to [Table 17–4](#) for field value descriptions.
5. Choose .



Exception Routing Rule Details (DEFAULT)

Exception Routing Sequence	<input type="text"/>	Routing Type	<input type="text"/>
Queue	<input type="text"/>	Java Class	<input type="text"/>
Routing Role	<input type="text"/>	Custom Role	<input type="text"/>
<input type="checkbox"/> Email Required		Email Template	<input type="text"/>
EmailId	<input type="text"/>		

Table 17–4 Exception Routing Rule Details Pop-Up Window

Field	Description
Exception Routing Sequence	Enter the exception routing sequence number.
Routing Type	<p>Select the appropriate type of routing rule. The valid value are:</p> <ul style="list-style-type: none"> • Use Department's Queue—Select this routing rule if you want to create the exception in the default queue of the department. • Use Specified Queue—Select this routing rule if you want to create the exception in the specified queue. • Use Interface to obtain Queue—Select this routing rule if you want to configure a java class for determining the queue in which the exception should be created. • Route to Role—If you select this type of routing rule, the SDF component picks up the seller or buyer organizations. If routing configuration exists for the organization the same rules are applied. • Route to Custom Role—If you select this type of routing rule, the SDF component picks up the custom organization. If routing configuration exists for the organization the same rules are applied. • Route to Exception Types Queue—If you select this routing rule, an exception is created in the default queue configured for the exception type.
Queue	Select the name of the alert queue to which the exceptions of this type are routed. For more information about creating alert queues, see Chapter 12, "Configuring Alert Queues" .
Java Class	If you have specified the routing type as "Use Interface to Obtain Queue", specify the java class which implements the interface.
Routing Role	Select the routing role. For example, BUYER, SELLER, and so forth.

Table 17–4 Exception Routing Rule Details Pop-Up Window

Field	Description
Custom Role	Enter the custom role.
Email Required	Check this box if you want to specify the e-mail id for this exception routing rule.
Email Template	Enter the template for the e-mails.
Email Id	Enter the e-mail id for one of the following routing rules: <ul style="list-style-type: none"> • Use Departments Queue • Use Specified Queue • Use Interface to Obtain Queue

17.3.4 Modifying an Exception Routing Rule

To modify an exception routing rule:

1. From the tree in the application rules side panel, choose Alert > Organization Exception Type Configuration. The Organization Exception Type for Organization window displays in the work area.
2. Select the applicable exception type and choose . The Exception Type for Organization Details window displays.
3. Select the applicable exception routing rule and choose . The Exception Routing Rule Details window displays.
4. Enter information in the applicable fields. Refer [Table 17–4](#) for field value descriptions.
5. Choose .

17.3.5 Deleting an Exception Routing Rule

To delete an exception routing role:

1. From the tree in the application rules side panel, choose Alert > Organization Exception Type Configuration. The Organization Exception Type for Organization window displays in the work area.
2. Select the applicable exception type and choose . The Exception Type for Organization Details window displays.
3. Select the applicable exception routing rule and choose .

17.3.6 Deleting an Exception Type for an Organization

To delete an exception type for an organization:

1. From the tree in the application rules side panel, choose Alert > Organization Exception Type Configuration.
2. The Organization Exception Type Configuration window displays with the list of Exception Type configured for an organization.
3. Select the exception type to be deleted. Choose .

Configuring Data Version Labels

Configuration data is an integral part of all implementations. Often, there is a need to track changes to an implementation's configuration. Furthermore, if the changes in configuration data are found to be inadequate, there is no easy way of rolling back changes to their original states.

In an offsite and onsite implementation model, master configuration data is maintained onsite, which is where the production environment is hosted. When a patch must be applied to the production environment, offsite test developers must write instructions regarding any configuration data changes for the patch and pass it to the onsite configuration manager; that is, certain values of business rules need to be changed. The onsite configuration manager has to replicate the configuration changes onto the production environment.

To make it easier to track versions of configuration data, or sets of changes to configuration data, Selling and Fulfillment Foundation includes the Configuration Data Versioning Tool, which is part of the Configuration Deployment Tool (CDT). It enables you to capture changes from a source database, compare and deploy them onto a target database (this can be the same or a different database).

The config table must have `AuditRequired` set to `Y` and the table name must exist in `config_db.xml`

Note: To enable this functionality the configuration table must have `AuditRequired` flag set to `Y`. By default, most of the configuration tables have `AuditRequired` flag set to `Y`.

You can create version labels in the Applications Manager to represent timestamps in a time line when changes occur in configuration data. The system can then identify any changes in the configuration data between timestamps of version labels based on the audit information in the system.

The Configuration Versioning Tool allows you to select different version labels from a source database and compare the data and apply them to a target database. You can see the details of each difference and detect conflicts. Once all conflicts are resolved, you can deploy the changes. For more information about this tool, see the *Selling and Fulfillment Foundation: Configuration Deployment Tool Guide*.

18.1 Creating Configuration Data Version Labels

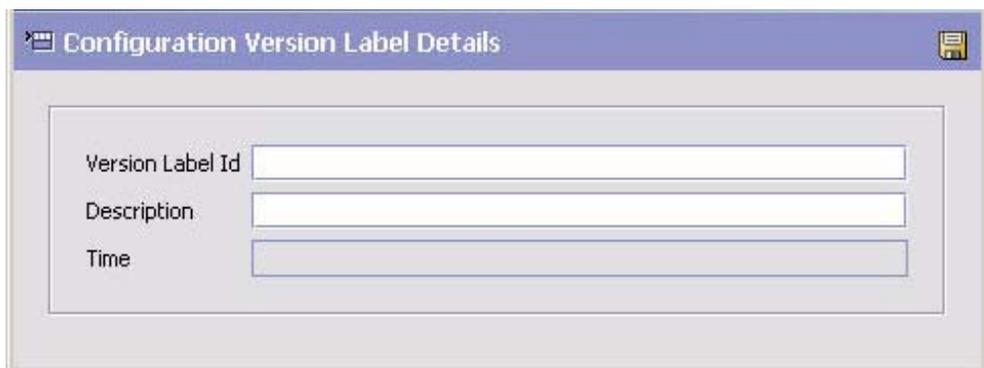
Before you can use the comparison tools, you must create version labels for the databases using the Applications Manager.

To create Configuration Version Labels:

1. In the Applications Manager, select Application Platform.
2. Select **Configuration Version Labels**.

The Configuration Version Labels screen is displayed on the right-hand panel. You can filter existing version labels based on Version Label ID and version label timestamps, and manage version labels based on the filter results.

3. To create or modify a new label, click  and enter a version label ID and description of the label.



The screenshot shows a window titled "Configuration Version Label Details". Inside the window, there are three input fields: "Version Label Id", "Description", and "Time". Each field is represented by a text label followed by a rectangular input box.

4. Click .

Table 18–1 Label Details Pop-up Window

Field	Description
Version Label ID	Enter the name of the identifier for this label.
Description	Enter a description for this label.
Time	The current timestamp displays by default for this label.

18.1.1 Modifying Configuration Data Version Labels

Once a Configuration Version Label has been created, it can be modified.

To modify a configuration version label:

1. In the Applications Manager, select Application Platform.
2. Select **Configuration Version Labels**.
The Configuration Version Labels screen is displayed on the right-hand panel.
3. Select the Version Label ID you want to modify and choose .
4. Enter information in the applicable fields. Refer [Table 18–1](#) for field value descriptions.
5. Choose .

18.1.2 Deleting Configuration Data Version Labels

To delete configuration version label:

1. In the Applications Manager, select Application Platform.
2. Select **Configuration Version Labels**.
The Configuration Version Labels screen is displayed on the right-hand panel.
3. Select the Version Label ID you want to delete.
4. Choose .

Configuring Qualified Tag Information

Qualified Tag Information branch is used to define Qualified Tag and Qualified Tag Types pertaining to rules and common codes.

QUALIFIED TAG

You can configure Qualified Tag to associate a rule or a common code to a particular Selling and Fulfillment Foundation version. You can define different rules and common codes based on the Qualified Tag. You can then associate a Qualified Tag to a particular Selling and Fulfillment Foundation version.

QUALIFIED TAG TYPE

Qualified Tag Type is used to configure the class that will validate the Qualified Tags against a particular Selling and Fulfillment Foundation version. You can associate a particular Qualified Tag Type with one or more Qualified Tags.

Note: If you configure two Qualifier Tags, Q1 and Q2, with Q1 being valid for Selling and Fulfillment Foundation versions V1 and V2, and Q2 being valid for Selling and Fulfillment Foundation versions V2 and V3, and you call the `getRuleDetails` API and pass V2 as the Selling and Fulfillment Foundation version, the system randomly returns one of the rule values.

You can use the Qualified Tag Information branch for:

- [Defining Qualified Tag Types](#)

- [Defining Qualified Tags](#)
- [Defining Qualified Tag Version Compatibility](#)

19.1 Defining Qualified Tag Types

You can specify Qualified Tag Type details here. You can create new Qualified Tag Type and classify them in one or more types.

Use Qualified Tag Information set-up for:

- [Creating a Qualified Tag Type](#)
- [Modifying a Qualified Tag Type](#)
- [Deleting a Qualified Tag Type](#)

19.1.1 Creating a Qualified Tag Type

To create a Qualified Tag type:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag Type. The Qualified Tag Type window displays in the work area.
2. Choose . The Qualified Tag Type Details window displays.
3. Enter information in the applicable fields. Refer to [Table 19–1](#) for field value descriptions.
4. Choose .

Figure 19–1 Qualified Tag Type Details Window

The screenshot shows a dialog box titled "Qualified Tag Type Details". It has a standard Windows-style title bar with a close button on the right. The main area contains three text input fields, each with a label to its left:

- Qualified Tag Type**: The first field, with the label in red text.
- Qualified Tag Type Description**: The second field.
- Qualified Tag Validator Class**: The third field.

Table 19–1 Qualified Tag Type Details Pop-Up Window

Field	Description
Qualified Tag Type	Enter name for the new Qualified Tag Type to be created. This is a mandatory field.
Qualified Tag Type Description	Enter a brief description of the Qualified Tag Type.
Qualified Tag Validator Class	Enter the Qualified Tag class name that validates the whether or not a Qualified Tag Id is valid for a particular application code and application version. Note: If left blank, default implementation of the Qualified Tag Validator class will be used to validate the Qualified Tag.

19.1.2 Modifying a Qualified Tag Type

Once a Qualified Tag Type has been created, it can be modified.

To modify an Qualified Tag Type:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag Type.
2. The Qualified Tag Type window displays with the list of Qualified Tag Types.
3. Select the Qualified Tag Type to be modified. Choose .

4. Enter information in the applicable fields. Refer [Table 19–1](#) for field value descriptions.
5. Choose .

19.1.3 Deleting a Qualified Tag Type

To delete an Qualified Tag Type:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag Type.
2. The Qualified Tag Type window displays with the list of Qualified Tag Types.
3. Select the Qualified Tag Type to be deleted. Choose .

19.2 Defining Qualified Tags

You can specify a list of Qualified Tags for rule and common code for a particular Selling and Fulfillment Foundation code and version. You can associate one or more Qualified Tags with a particular Qualified Tag Type.

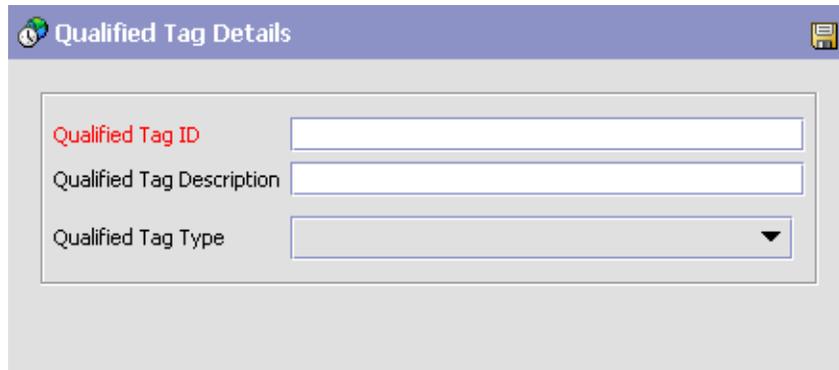
Use Qualified Tag Information set-up for:

- [Creating a Qualified Tag](#)
- [Modifying a Qualified Tag](#)
- [Deleting a Qualified Tag](#)

19.2.1 Creating a Qualified Tag

To create a Qualified Tag:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag. The Qualified Tag window displays in the work area.
2. Choose . The Qualified Tag Details window displays.
3. Enter information in the applicable fields. Refer to [Table 19–2](#) for field value descriptions.
4. Choose .



The screenshot shows a window titled "Qualified Tag Details". It contains three input fields: "Qualified Tag ID" (a text box), "Qualified Tag Description" (a text box), and "Qualified Tag Type" (a dropdown menu).

Table 19–2 Qualified Tag Details Pop-Up Window

Field	Description
Qualified Tag ID	Enter the Qualified Tag ID. This is a mandatory field.
Qualified Tag Description	Enter a description about the Qualified Tag.
Qualified Tag Type	Select the Qualified Tag Type that you want to associate with the Qualified Tag Id. This is a mandatory field.

19.2.2 Modifying a Qualified Tag

Once a Qualified Tag has been created, it can be modified.

To modify a Qualified Tag:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag.
2. The Qualified Tag window displays with the list of Qualified Tags.
3. Select the Qualified Tag to be modified. Choose .
4. Enter information in the applicable fields. Refer [Table 19–2](#) for field value descriptions.
5. Choose .

19.2.3 Deleting a Qualified Tag

To delete a Qualified Tag:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag.
2. The Qualified Tag window displays with the list of Qualified Tags.
3. Select the Qualified Tag to be deleted. Choose .

19.3 Defining Qualified Tag Version Compatibility

Using Qualified Tag Version Compatibility, you can associate Selling and Fulfillment Foundation code and version with a particular Qualified Tag.

Use Qualified Tag Information set-up for:

- [Configuring Qualified Tag Version Compatibility](#)
- [Modifying Qualified Tag Version Compatibility](#)
- [Deleting Qualified Tag Version Compatibility](#)

19.3.1 Configuring Qualified Tag Version Compatibility

To configure Qualified Tag version compatibility:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag Version Compatibility. The Qualified Tag Version Compatibility window displays in the work area.
2. Choose . The Qualified Tag Version Compatibility Details window displays.
3. Enter information in the applicable fields. Refer to [Table 19–3](#) for field value descriptions.
4. Choose .

Table 19–3 Qualified Tag Version Compatibility Details Pop-Up Window

Field	Description
Application Code	Select the Selling and Fulfillment Foundation code. This is a mandatory field.
Qualified Tag ID	Select the Qualified Tag ID. This is a mandatory field.
Minimum Version Supported	Select the minimum Selling and Fulfillment Foundation version which supports the Qualified Tag identifier. Note: This is a mandatory field, if you have not specified the Maximum Version Supported.
Maximum Version Supported	Select the maximum Selling and Fulfillment Foundation version which supports the Qualified Tag identifier. Note: This is a mandatory field, if you have not specified the Minimum Version Supported.
Support Level	Enter the level of support for the Selling and Fulfillment Foundation version which is below the supported minimum version.

19.3.2 Modifying Qualified Tag Version Compatibility

Once the Qualified Tag version compatibility has been configured, it can be modified.

To modify Qualified Tag version compatibility:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag Version Compatibility.
2. The Qualified Tag Version Compatibility window displays with the Qualified Tag Version Compatibility list containing Application Code, Qualified Tag Id, Minimum Version Supported, Maximum Version Supported, and Support Level fields.
3. Select the Qualified Tag Version Compatibility list item to be modified. Choose .
4. Enter information in the applicable fields. Refer [Table 19–3](#) for field value descriptions.
5. Choose .

19.3.3 Deleting Qualified Tag Version Compatibility

To delete a Qualified Tag version compatibility:

1. From the tree in the application rules side panel, choose Qualified Tag Information > Qualified Tag Version Compatibility.
2. The Qualified Tag Version Compatibility window displays with the Qualified Tag Version Compatibility list containing Application Code, Qualified Tag Id, Minimum Version Supported, Maximum Version Supported, and Support Level fields.
3. Select the Qualified Tag Version Compatibility list item to be deleted. Choose .

Configuring Attribute Postfix Rules

Defining common codes for attribute postfixes enables you to set up postfixes to associate with attributes. Defining postfixes ensures that each customer sees attribute data in a familiar format. For example, you can define "dollars" as the postfix for the price attribute.

20.1 Defining Attribute Postfix Codes

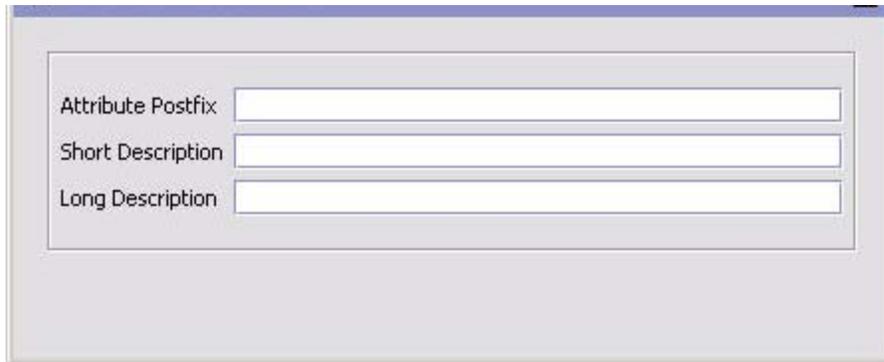
You can set up common codes for attribute postfixes. You can use the Attribute Postfix branch for:

- [Creating an Attribute Postfix Definition](#)
- [Modifying an Attribute Postfix Definition](#)
- [Deleting an Attribute Postfix Definition](#)

20.1.1 Creating an Attribute Postfix Definition

To create an attribute postfix definition:

1. From the tree in the application rules side panel, choose Attribute Postfix. The Attribute Postfix window displays in the work area.
2. Choose . The Attribute Postfix Details pop-up window displays.



The image shows a dialog box titled 'Attribute Postfix Details'. It contains three text input fields:

- Attribute Postfix
- Short Description
- Long Description

3. In Attribute Postfix, enter an attribute postfix definition.
4. In Short Description, enter a brief description of the attribute postfix.
5. In Long Description, enter a more detailed description of the attribute postfix.
6. Choose .

20.1.2 Modifying an Attribute Postfix Definition

To modify an attribute postfix definition:

1. From the tree in the application rules side panel, choose Attribute Postfix. The Attribute Postfix window displays in the work area.
2. Select the applicable attribute postfix definition and choose . The Attribute Postfix Details pop-up window displays.
3. In Short Description, enter a brief description of the attribute postfix definition.
4. In Long Description, enter a more detailed description of the attribute postfix definition.
5. Choose .

20.1.3 Deleting an Attribute Postfix Definition

To delete an attribute postfix definition:

1. From the tree in the application rules side panel, choose Attribute Postfix. The Attribute Postfix window displays in the work area.
2. Select the applicable attribute postfix definition and choose .

Configuring Analytics

You can use Analytics to associate pre-existing region schemas with Data Warehouse Analytics. This association enables reporting on the best region matching mechanism for features such as sourcing and resource pools. In analytics, the region schema determines the level of granularity that is used in Data Warehouse reports. For more information about configuring region schemas, see the [Chapter 13, "Configuring Region Definitions"](#).

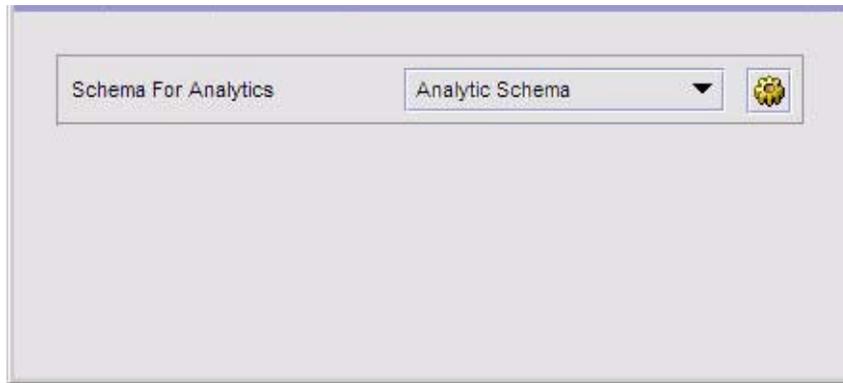
Note: When associating a region schema with analytics, do not specify a schema that includes 9-digit zip codes. Otherwise, system performance is effected.

21.1 Defining Analytics Region Usage

Use the Analytics branch to define a region schema for analytics.

To define analytics region usage:

1. From the tree in the application rules side panel, choose Analytics > Region Usage For Analytics. The Region Usage For Analytics window displays.



2. From Schema for Analytics, select the region schema you want to use for Data Warehouse Reports.
3. Choose .

Time-Triggered Transaction Reference

Selling and Fulfillment Foundation 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`. To override this property, add an entry in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

For more information about statistics persistence, see the *Selling and Fulfillment Foundation: 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 is 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.

Selling and Fulfillment Foundation 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

Selling and Fulfillment Foundation 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 run successfully in a given time interval.
- `ExecuteMessageError` - The number of jobs that failed to run 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 8.5 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Selling and Fulfillment Foundation.

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 more information about running time-triggered transactions, see the *Selling and Fulfillment Foundation: Installation Guide*. For more information about fine-tuning system performance while running them concurrently, see the *Selling and Fulfillment Foundation: Performance Management Guide*.

A.1.1 Steps to Complete Before Scheduling Time-Triggered Transactions

Before running and scheduling a time-triggered transaction, ensure that you have completed the following:

1. Configure a JMS Connection Factory to correlate with the QCF name configured for the time-triggered transaction. The Selling and Fulfillment Foundation factory defaults include the `AGENT_QCF` as the JMS Connection Factory. For more information about configuring JMS, see the documentation for your specific application server.
2. Configure JMS Server Destinations to correlate with the group or individual name of the time-triggered transaction. The Selling and Fulfillment Foundation factory defaults include the `DefaultAgentQueue` as the server destination.

Note: Do not put a dot (.) in the name of a JMS Server Destination, for example, 'A.0001'. If you do, Selling and Fulfillment Foundation is unable to communicate with it.

3. Using the Applications Manager, configure each time-triggered transaction required for your business process as described in [Section 4.2.6, "Defining Transactions"](#). Each set of time-triggered transaction criteria parameters must ensure the appropriate association of a JMS Agent Server.

A.2 Configuring Communication Between an Agent and a JMS Server

Setting up communication between an agent (time-triggered transaction) and a remote JMS server requires that you do some prerequisite setup on your JMS system, then do some configuration within the application, which consists of the following procedures:

- If an initial context factory code for your JMS system is not provided with the application, you must create one. See "Create an Initial Context Factory Code" on page 499 for the list of codes that are provided.
- Defining the transaction details – the time-triggered transaction, or agent, must be edited to include connection information for your JMS system and the initial context factory you create. See [Section A.2.3, "Define the Transaction Information"](#).

For more information about time-triggered transactions and how they fit into the larger picture of application business process modeling, see the *Configuring Process Models* chapter. Also see the *Configuring Alert Queues* chapter for additional information about queues and agents.

A.2.1 Prerequisites

Before starting, complete these tasks for your JMS Server. See your JMS Server documentation for more information about performing these tasks.

1. Configure the JMS Queue Connection Factory (QCF) and queues on your JMS server.
2. Configure the JNDI representation of the queues on your JMS server.

Ensure that you have the following information available from these tasks:

- JNDI name for each queue
- JNDI QCF lookup
- JMS location - the provider URL for the JMS server

Once you have completed the preceding tasks, complete the next two procedures in the order shown. These are both done in the application.

A.2.2 Create an Initial Context Factory Code

Using an Initial Context Factory (ICF) class enables remote Java clients to connect to your application. This class is provided by the application vendor. The application uses ICF codes to identify these when setting up agents. Initial context factory codes are predefined in the application for the following JMS vendors:

- WebSphere MQ (for MQSeries accessed through a WebSphere iiop URL)
- File (for MQSeries accessed through a file URL, as with WebLogic)
- WebLogic (for WebLogic JMS)
- Jboss (for JBoss JMS)

If you are using a JMS server that is **not** in the preceding list (for example, ActiveMQ), you must create an initial context factory code for it in the application:

1. Open the Applications Manager. From the tree in the application rules side panel, choose System Administration > Initial Context Factory Codes. The Initial Context Factory Codes window displays in the work area.
2. Select the + icon to create a new initial context factory code. The Initial Context Factory window is displayed.
3. In the Initial Context Factory field, enter the name of the class provided by your JMS vendor. For example, for ActiveMQ, the class name is `org.apache.activemq.jndi.ActiveMQInitialContextFactory`.
4. In the Short Description field, enter a descriptive name, up to 40 characters. Make note of this name, because you will use it in the next procedure (see [Section A.2.3, "Define the Transaction Information"](#)). For ActiveMQ, enter **ActiveMQ**.
5. In the Long Description field, enter a more detailed description for the initial context factory, up to 100 characters.
6. Save the new initial context factory code and close the window.

For more information about ICFs, see *Creating an Initial Context Factory Code*.

A.2.3 Define the Transaction Information

For the JMS server to communicate with the application, there must be a time-triggered transaction configured with the JMS server and ICF information.

1. Open the Applications Manager. From the tree in the application rules side panel, double-click Process Modeling. The Process Modeling window displays in the work area.
2. Select the desired tab, then Base Document Type, then double-click Process Type.
3. Double-click the transaction that corresponds to the agent to be run.
4. Select the Time Triggered tab.
5. Create or select an existing Agent Criteria Definition to edit.
6. The Agent Criteria Details screen is displayed. Select the Runtime Properties tab.
7. Select an existing Agent Server from the list or create your own (recommended).
8. Select an existing Alert Queue from the list or create your own.
9. In the JMS Queue Name field, enter the JNDI name for the queue that you created. See [Section A.2.1, "Prerequisites"](#).
10. Enter the desired number of threads the agent should run (recommended not to exceed 5 threads - if more than 5 are needed, start another agent in its own JVM).
11. Select the Initial Context Factory code you created. See [Section A.2.2, "Create an Initial Context Factory Code"](#).
12. In the QCF Lookup field, enter the JNDI QCF lookup for the queue that you created (this is the Queue Connection Factory created for the applicable JMS Server). See [Section A.2.1, "Prerequisites"](#).
13. Enter the Provider URL. This is the location where the JMS system resides, and is JMS vendor specific.
14. Select whether the agent should trigger itself (recommended) and at what interval (in minutes) or use an external trigger (triggeragent.sh in the `<install_dir>/install/bin` directory).

15. See Setting up the JMS Security Properties for information about setting the JMS Security option.
16. Leave the Criteria Parameters tab values at the default values.
17. Save the Agent Criteria Details and close the window.
18. Launch the agent in its own JVM by executing the `startagentserver.sh/cmd` script in the `<install_dir>/install/bin` directory.

For additional information about defining transactions and about this procedure, see the sections, *Defining Transactions* and *Specifying a Transaction as Time-Triggered*.

A.3 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 8.5 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Selling and Fulfillment Foundation.

Note: All Business Process Time-Triggered Transactions have a `CollectPendingJobs` criteria parameter. If this parameter is set to `N`, the agent does not collect information about the pending jobs pertaining to this monitor. This pending job information is used for monitoring the monitor in the System Management Console.

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 Asynchronous Request Processor

This transaction completes any API request or service request in offline mode. It picks up the API messages or service messages from the YFS_ASYNC_REQ table and invokes the corresponding API or service. The messages can be inserted into the YFS_ASYNC_REQ table using the createAsyncRequest API. Some of the business transactions in the Sterling Warehouse Management System also insert the messages into the YFS_ASYNC_REQ table.

Attributes

Following are the attributes for this time-triggered transaction:

Table A–1 Asynchronous Request Processor Attributes

Attribute	Value
Base Transaction ID	ASYNC_REQ_PROCESSOR
Base Process Type	General
Abstract Transaction	No

Criteria Parameters

Following are the criteria parameters for this transaction:

Table A–2 Asynchronous Request Processor 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.
Lead Days	Number of days before the present date the agent will purge the records. If left blank or specified as 0 (zero), it defaults to 30.

Table A–2 Asynchronous Request Processor Parameters

Parameter	Description
Maximum Error Count	Maximum number of times the record is processed if an exception is thrown. Once the number of unsuccessful attempts equals this number, that record is not processed further by the agent. If left blank or specified as 0 (zero), it defaults to 20.
Reprocess Interval In Minutes	Time in minutes after which the transaction will be reprocessed - after it has been processed and has thrown an exception.
ColonyID	Required in a multischema deployment where the YFS_ASYNC_REQ table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None

Pending Job Count

None

Events Raised

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

Table A–3 Events Raised by the Asynchronous Request Processor

Transaction/Event	Key Data	Data Published*	Template Support?
HAS_EXCEPTIONS	None	YCP_ASYNC_REQ_PROCESSOR.HAS_EXCEPTIONS.html	Yes

*These files are located in the following directory:
 <INSTALL_DIR>/xapidocs/api_javadocs/XSD/HTML

A.3.2 Case Insensitive Data Loader

The Case Insensitive Data Loader agent migrates data from columns marked CaseInsensitiveSearch to shadow columns. The agent uses the transaction criteria to identify the records that need to be updated and then converts the original column values to lowercase values in the shadow columns. For more information about enabling case insensitive searches, refer to the *Selling and Fulfillment Foundation: Extending the Database Guide*.

Attributes

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

Table A-4 Case Insensitive Data Loader Attributes

Attribute	Value
Base Transaction ID	DATA_LOADER
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–5 Case Insensitive Data Loader 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. <ul style="list-style-type: none"> • If left blank or the number specified is less than 10000, it defaults to 5000. • If the number specified is greater than 10000, then that value is used.
CollectPendingJobs	If this parameter is set to "N", the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
TableType	Required in a multischema deployment when a table may exist in multiple schemas. Valid Values: CONFIGURATION, TRANSACTION, MASTER. If set to CONFIGURATION, the agent runs for the records associated with tables that have TableType as CONFIGURATION. If set to TRANSACTION, the agent runs for the records associated with tables that have TableType as TRANSACTION.
Table Name	Required. The table name for the records to be migrated to shadow columns.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

None.

Events Raised

None.

A.3.3 Change Load Status

This transaction is equivalent to the `changeLoadStatus()` API. For detailed information about this transaction, see the *Selling and Fulfillment Foundation: 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-6 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	<code>changeLoadStatus()</code>

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–7 *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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–8 *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 (\leq) the CurrentDate value in the YFS_Task_Q table.

Events Raised

This transaction raises events as specified under the `changeLoadStatus()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.4 Change Shipment Status

This transaction is equivalent to the `changeShipmentStatus()` API. For detailed information about this transaction, see the *Selling and Fulfillment Foundation: 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–9 *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–10 *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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–11 *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 (\leq) the current date value in the YFS_Task_Q table.

Events Raised

This transaction raises events as specified under the `changeShipmentStatus()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.5 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.4.3.5, "Delivery Plan Purge"](#)).

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 Console must schedule purge jobs.

Attributes

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

Table A–12 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–13 Close Delivery Plan Criteria Parameters

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.

Table A–13 Close Delivery Plan 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–14 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–15 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.3.6 Close Load

To boost system performance, this transaction serves as a temporary purge until the Load Purge deletes load-related data (see [Section A.4.3.13, "Load Purge"](#)).

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–16 *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-17 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-18 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-19 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.3.7 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_ysc_ext.in` 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 the value specified for the `yfs.closemanifest.online` property in the `yfs.properties_ysc_ext.in` file. To override this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

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–20 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–21 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 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 Application 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following are statistics are tracked for this transaction:

Table A–22 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–23 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.3.8 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 for all the order lines reaches the configured pickup status. 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: The Close Order agent must be run before running the Monitor agent in order to avoid alerts getting raised for cancelled orders.

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-24 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–25 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–26 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–27 Events Raised by the Close Order Transaction

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

A.3.9 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–28 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, Sales 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–29 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.
Node	Mandatory. Node for which the Close Receipts needs to be run.
AgentCriteriaGroup	Used to classify nodes. This value can be accepted by WMS time-triggered transactions that 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 Application Platform > System Administration > Agent Criteria Groups.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–30 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-31 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" displays. 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 *Sterling 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.3.10 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.4.3.31, "Shipment Purge"](#)).

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–32 *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–33 *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.

Table A–33 Close Shipment Criteria Parameters

Parameter	Description
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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following are statistics are tracked for this transaction:

Table A–34 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–35 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.3.11 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–36 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–37 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.

Table A–37 Collect Shipment Statistics Criteria Parameters

Parameter	Description
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that 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 Application Platform > System Administration > Agent Criteria Groups.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–38 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–39 Events Raised by the Collect Shipment Statistics Transaction

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

A.3.12 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-40 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-41 Consolidate Additional 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 inventory item records (whose additional supplies and demands are consolidated_ to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
ColonyID	Required in a multischema deployment where the YFS_INVENTORY_SUPPLY_ADDNL and YFS_INVENTORY_DEMAND_ADDNL tables may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-42 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.3.13 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 information, see the details provided under the `createShipment()`, `changeShipment()`, and `releaseOrder()` APIs in the *Selling and Fulfillment Foundation: 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 are never consolidated with any other release.

Attributes

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

Table A-43 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–43 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</code>. • <code>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>. For more information, see the <i>Selling and Fulfillment Foundation: Javadocs</i>.

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–44 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Pending Job Count

Table A-45 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-46 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 as specified under the `createShipment()` and `changeShipment()` APIs in the *Selling and Fulfillment Foundation: Javadocs*.

However, note that the template name would read `<TransactionId>.ON_SUCCESS.xml`. The XML and DTD depicted above represent the output that the abstract transaction CONSOLIDATE_TO_SHIPMENT transaction is capable of generating.

A.3.14 Create Catalog Index

The Create Catalog Index transaction builds the Apache Lucene index file that is used by catalog search. This index file enhances search performance by storing denormalized item data that has been extracted from the Selling and Fulfillment Foundation database.

The Create Catalog Index transaction can be configured to perform the following tasks:

- Run either a scheduled index build or user-initiated index build
- Build either a full or incremental index file
- Activate the index file

The Index Building Process

The Create Catalog Index transaction provides an agent for index building. Index building is a multi-thread process in which the index building agent extracts item and item-related information from the active selling catalog in the Selling and Fulfillment Foundation database and writes this information to multiple files. The files identify the item data that should be included in the final index. After the agent finishes writing the files, it merges them into the final index file.

The multi-thread process provides the advantage of parallel processing. Large amounts of database data are segmented and processed simultaneously, which is faster and more scalable than sequentially processing one long file.

When writing information to multiple files, the index building agent performs the following tasks for each item before looping to the next item:

- Queries the Selling and Fulfillment Foundation database for data about the item.
- Uses information from the XML configuration file and extension file to determine the data that be retrieved from the query.
- Retrieves relevant data from the Selling and Fulfillment Foundation database.
- Creates a Lucene document for the item.

After the transaction creates a Lucene document for each item, the transaction writes the documents to the index file based on the organization and the organization's locales.

Attributes

[Table A-47](#) displays the attributes for the Create Catalog Index transaction.

Table A-47 Create Catalog Index

Attribute	Value
Base Transaction ID	Create_Catalog_Index
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YCMParseAssetUE

Criteria Parameters

[Table A-48](#) displays the criteria parameters for the Create Catalog Index transaction.

Table A-48 Create Catalog Index

Parameter	Description
Organization Code	Required. The organization code of the catalog organization or subcatalog organization that maintains the search index.
Number of Messages	Required. Number of messages to use when building the index file. Selling and Fulfillment Foundation processes only one message per thread. For example, if Number of Messages is set to 10 and Threads is set to 3, Selling and Fulfillment Foundation processes only 3 messages at a time. For more information about fine-tuning system performance, see the <i>Selling and Fulfillment Foundation: Performance Management Guide</i> .

Table A-48 Create Catalog Index

Parameter	Description
Incremental Build	<p>Y or N.</p> <p>Y to rebuild the existing index file. If you specify Y, Selling and Fulfillment Foundation rebuilds the index based on the last successful index build.</p> <p>N to build a full index file.</p> <p>This parameter is ignored for user-initiated index builds. However, if scheduled builds are configured, ensure that you specify whether you want a full or incremental index build.</p>
Category Domain	<p>Optional. The catalog from which the index is built. The active selling catalog of the catalog organization or subcatalog organization is the default. If scheduled builds are configured, ensure that you specify a catalog.</p>
Auto Activate	<p>Y or N. Optional.</p> <p>Y to activate the index after building the index file.</p> <p>The default is N.</p>

Table A–48 Create Catalog Index

Parameter	Description
Auto Insert Search Index Trigger	<p>Y or N. Optional.</p> <p>Y to enable scheduled builds of the catalog index file. The agent refers to information stored in the YFS_SEARCH_INDEX_TRIGGER table to determine when to run the scheduled index build. Specify the type of index build, whether full or incremental, in the agent criteria.</p> <p>N to enable user-initiated builds of the catalog index file. The agent continuously queries the YFS_SEARCH_INDEX_TRIGGER table to determine whether an index build is indicated. If a user starts an index build from the Business Center, the status setting in the table changes to Scheduled, triggering the agent to build the index. The user specifies the type of index build, whether full or incremental, from the Business Center.</p> <p>After a scheduled or user-initiated build runs, the user can activate the index from the Business Center. Alternatively, the agent can be configured to automatically activate the index.</p> <p>To allow both scheduled and user-initiated index builds, configure the transaction to include two instances of the agent. Configure one instance to trigger user-initiated builds and the second instance to trigger scheduled index builds.</p>
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

Table A–49 shows the statistics that are tracked for the Create Catalog Index transaction.

Table A–49 Create Catalog Index

Statistic Name	Description
SearchIndicesBuilt	Number of search indices that have been built.

Pending Job Count

None.

Events Raised

None.

A.3.15 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 about the creation of chained orders, see the information provided under the `createChainedOrder()` API and the `YFSDetermineChainedOrderForConsolidation` user exit in the *Selling and Fulfillment Foundation: Javadocs*.

This transaction should be invoked after order scheduling.

Attributes

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

Table A–50 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–51 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

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-52 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 (<=) the current date value in the YFS_Task_Q table.

Events Raised

This transaction raises events as specified under the createChainedOrder() API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.16 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 about the creation of derived orders, see the details provided under the createDerivedOrder() API and YFSDetermineDerivedOrderForConsolidation user exit in the *Selling and Fulfillment Foundation: Javadocs*.

Attributes

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

Table A-53 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-54 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–55 *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 as specified under the `createDerivedOrder()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.17 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.3.18, "Create Shipment Invoice"](#).

Attributes

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

Table A–56 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–57 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–58 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 as specified under the `createOrderInvoice()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.18 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.3.17, "Create Order Invoice"](#).

Attributes

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

Table A–59 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–60 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–61 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 as specified under the `createShipmentInvoice()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.19 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-62 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	getNodeMinimumNotificationTime

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A-63 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.

Table A–63 ESP Evaluator Criteria Parameters

Parameter	Description
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that 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 Application Platform > System Administration > Agent Criteria Groups.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

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–64 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.3.20 Item Based Allocation

The Item Based Allocation transaction allocates unpromised and promised demands of existing orders to more suitable supplies based upon inventory items and nodes which have been triggered for the Item Based Allocation process in the YFS_IBA_TRIGGER table.

The Item Based Allocation agent obtains and processes all Item Based Allocation triggers from the YFS_IBA_TRIGGER table that meet the following conditions:

- IBA_RUN_REQUIRED = "Y"
- LAST_IBA_PROCESSED_TS was 'x' hours before current time, where 'x' is from the 'Item Based Allocation Agent Execution Interval (in hours)' rule in the Installation rules. For more information about installation rules, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*. This rule is used to indicate the interval that the Item Based Allocation agent should not reprocess the triggers in the YFS_IBA_TRIGGER table, which were processed earlier. This prevents the IBA agent from over-processing the item and node combination in the given time interval to avoid any high loads on the system.
- PROCESSING_BY_AGENT="N" or PROCESS_OVER_BY_TS is before the current timestamp. The PROCESSING_BY_AGENT field is used to prevent the picking up of the IBA trigger which is being processed by another instance of the agent.

If InventoryOrganizationCode is specified in the agent criteria, only the IBA trigger with inventory items of that inventory organization is retrieved.

For each triggered item and node combination, the agent finds all of the applicable order lines or order line reservations that contain the item and node and tries to move their unpromised and promised demands to more suitable available supplies based on user-configured IBA selection rules or FIFO (First-In-First-Out) IBA selection rules.

Selling and Fulfillment Foundation creates new positive order line reservations with the matched supply's first ship date and negative order line reservations for the existing demand ship date. Once all orders are processed, they are placed on hold to be rescheduled if changes are detected in the order line reservations.

Note: The following configuration is required for the Item Based Allocation process:

- The Use Item Based Allocation rule needs to be enabled.
 - Item and node need to have Item Based Allocation Allowed enabled.
 - A hold type is required to be set up for the change order line reservations modification type so that the order can be placed on hold for rescheduling. For more information, refer to the *Selling and Fulfillment Foundation: Javadocs*.
-
-

Note: The 'When a line is backordered, backorder against the highest priority ship node' rule should be checked in order to reallocate backordered demand. For more information, see the Fulfillment Rules section in the *Sterling Distributed Order Management: Configuration Guide*.

Before processing the Item Based Allocation logic, the Item Based Allocation agent updates the following fields on the Item Based Allocation trigger:

- PROCESSING_BY_AGENT = "Y". This indicates that an instance of the agent is currently processing this trigger.
- PROCESS_OVER_BY_TS = current time + 1 hr. This indicates the expected time that the agent should finish with processing this IBA trigger. One hour is the fixed window and cannot be changed. Selling and Fulfillment Foundation treats the PROCESSING_BY_AGENT flag as "N" regardless of the actual value when current timestamp is after this timestamp.
- IBA_RUN_REQUIRED = "N". This resets the IBA_RUN_REQUIRED flag back to "N".

Obtaining a List of Demands Based on Applicable Order Release Statuses and Order Line Reservations to be Allocated

A list of demands is derived from applicable order release statuses and order line reservations, which have the item and node in the IBA trigger. The following types of demands are retrieved:

- Demands of chained orders
- Demands of orders with chained order already created
- Demands of orders with procurement node but chained order creation is not yet created
- Demands of orders without procurement node
- Demands from order line reservations

The demand quantity is derived based on the order release status quantity with the status from the Status Inventory Type configuration that has a demand type, which considers the supply type with 'Use Consider Demand Type for Item Based Allocation' enabled. For more information, refer to the *Sterling Global Inventory Visibility: Configuration Guide*.

Obtaining a List of Available Supplies for Allocation

Selling and Fulfillment Foundation obtains the available supply based on the availability of the item at the node by ignoring unpromised and promised demands. If the inventory organization maintains its inventory externally, the external availability can be read by the `YFSGetExternalInventoryUE` user exit. Only the availability of supplies that consider the 'Demand Type Look for Availability during Item Based Allocation' are used in the allocation logic. For more information, refer to the *Sterling Global Inventory Visibility: Configuration Guide*.

Note: Allocated demands should be matched with the same supplies as "Demand to look for during release".

Matching Demands Against Supplies in FIFO (First-In-First-Out) Order

Selling and Fulfillment Foundation sorts the list of available supplies in the order of the first shippable date (ETA), and matches the obtained list of demands using the top-down logic (unlike the normal matching logic for obtaining availability, where matches are based on the closest ETA). Demands are allocated in the following orders:

- Demands of chained orders - first based on user-configured sequencing rules, and then in ascending order of order creation date. (These types of demands are matched based on the closest ETA to avoid any changes in the chained orders).
- Demands of orders with a chained order already created - first based on user-configured sequencing rules, then in ascending order of product availability date. (These types of demands are matched based on the closest ETA to avoid any changes in the orders).
- Demands of orders for which procurement node and chained order creation is imminent (within the advanced notification time window) - first based on user-configured sequencing rules, then in order of order creation date.
- Demands of orders without a procurement node and within the release window (advanced notification time window) - first based on user-configured sequencing rules, then in order of order creation date.
- Demands from order line reservations on the order lines in the order of requested reservation date, and left-over demands (outside of the advanced notification time window) of orders with or without a procurement node, first based on user-configured sequencing rules and then in the order of order creation date.
- Demands from inventory reservations in the order of ship date.

Notice that different types of demands are given different priorities based on their significance. The demands of chained orders or orders related to chained orders are treated with a higher priority than the demands of normal orders. Furthermore, the demands with a ship date within the advanced notification time window also have a higher priority than the demands with a date outside of the advanced notification time window.

Updating Order Reservations for the Matched Demands

After matching the available supply and demand in user-configured sequencing and then in FIFO order, the system builds up a list of order line reservation changes and inventory demand changes (corresponding to the order line reservation changes) and summarize them to optimize the number of order reservation updates and inventory updates. Negative order line reservations are added for the matched demands. Positive order reservations are added for the matched demands with the product availability date set to the matched supplies' first ship date.

After the Item Based Allocation agent completes its tasks for an Item Based Allocation trigger, it updates the fields of the trigger with the following values:

- IBA_REQUIRED = "N"
- LAST_IBA_PROCESSED_TS = current timestamp.
- PROCESS_OVER_BY_TS = current timestamp.
- PROCESSING_BY_AGENT = "N"

The Item Based Allocation agent should be used in conjunction with the rescheduling process as the rescheduling process reschedules the affected orders by utilizing the order line reservations created by the Item Based Allocation process.

Attributes

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

Table A-65 Item Based Allocation Attributes

Attribute	Value
Base Transaction ID	ITEM_BASED_ALLOCATION
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	changeOrder – for updating the order line reservations created as part of the Item Based Allocation process.
User Exits Called	None

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A-66 Item Based Allocation 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	The inventory organization code of the inventory items which are processed by the Item Based Allocation agent. If provided, only the IBA triggers with the inventory item that belongs to this inventory organization are processed.
ColonyID	Required in a multischema deployment where the YFS_IBA_TRIGGER table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–67 *Item Based Allocation Statistics*

Statistic Name	Description
NumOrdersProcessed	Number of orders processed by the Item Based Allocation agent.
NumOrdersRequiredReschedule	Number of orders required rescheduling as the result of Item Based Allocation process.

Pending Job Count

None.

Events Raised

This transaction raises events as specified under the changeOrder API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.21 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–68 *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

Table A–68 Mark Load As Trailer Loaded Attributes

Attribute	Value
APIs Called	None
User Exits Called	None

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–69 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–70 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.3.22 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-71 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-72 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.
ColonyID	Required in a multischema deployment where the YFS_INVENTORY_SHIPMENT, YFS_INVENTORY_RECEIPT, and the YFS_INVENTORY_MATCH tables may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-73 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.3.23 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.3.24, "Payment Execution"](#).

Attributes

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

Table A–74 *Payment Collection Attributes for Sales Orders*

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

Table A–75 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–76 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. The enterprise for which the transaction needs to be run. If left blank, orders for all enterprises are processed. If specified, only orders for that enterprise are processed.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-77 Payment Collection Statistics

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

Pending Job Count

For this transaction the pending job count is the number of orders in the appropriate payment statuses with the value of the AUTHORIZATION_EXPIRATION_DATE is less than or equal to (\leq) the current date. 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-78 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
REQUEST_PAYMENT_STATUS		YFS_PAYMENT_COLLECTION.REQUEST_PAYMENT_STATUS.xml	Yes
ON_LIABILITY_TRANSFER	modifyOrder_dbd.txt	YFS_PAYMENT_COLLECTION.ON_LIABILITY_TRANSFER.xml	Yes
ON_INVOICE_COLLECTION	order_dbd/txt	YFS_CREATE_ORDER_INVOICE.ON_INVOICE_COLLECTION.xml	Yes

A.3.24 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–79 *Payment Execution Attributes for Sales Orders*

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

Table A–80 *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–81 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
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–82 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.

Table A–82 Payment Execution Statistics

Statistic Name	Description
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–83 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 as specified under the `executeCollection()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.25 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–84 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–85 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.
ColonyID	Required in a multischema deployment where the YFS_INVENTORY_MATCH table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–86 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–87 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.3.26 Process Order Hold Type

You can create a time-triggered transaction, derived from the PROCESS_ORDER_HOLD_TYPE abstract transaction. 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 the YFS_TASK_Q table 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 no hold types can be processed, the YFS_TASK_Q record is deleted. If some hold types can be processed, the processOrderHoldType user exit is invoked with the list of hold types to be processed. The processOrderHoldType user exit returns the list of hold types that can be removed from the order.

The transaction then modifies the order and updates the order hold type list based on the output returned by the processOrderHoldType user exit. If now no hold types can be processed, the YFS_TASK_Q record is deleted. If some hold types can still be processed, YFS_TASK_Q is updated with the next available date.

You can also call the processOrderHoldType user exit to add new hold types or change the status of a hold type that is already applied to an order. For more information about the processOrderHoldType user exit, see the *Selling and Fulfillment Foundation: Javadocs*.

Attributes

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

Table A–88 Process Order Hold Type Attributes

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

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–89 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.

Table A–89 Process Order Hold Type Parameters

Parameter	Description
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.
ColonyID	Required in a multischema deployment where the YFS_TASK_Q table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

None

Events Raised

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

Table A–90 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

Table A–90 Events Raised by Process Order Hold Type Transaction

Transaction/Event	Raised when...	Key Data	Data Published	Template Support?
ON_ORDER_LINE_HOLD_TYPE_STATUS_CHANGE	The status of a hold type is changed.	modifyOrder_dbd.txt	YFS_ON_ORDER_LINE_HOLD_TYPE_STATUS_CHANGE.xml	Yes
<p>* Note: Some of the elements and attributes are not template-driven. Refer to the xml for element level details.</p>				

A.3.27 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–91 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-92 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

None

Events Raised

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

Table A-93 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.3.28 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-94 Publish Negotiation Results Attributes

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

Table A–94 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–95 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Pending Job Count

Table A–96 Publish Negotiation Results Statistics

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

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

value less than or equal to (\leq) the current date value in the YFS_Task_Q table.

Events Raised

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

Table A-97 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.3.29 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 information provided under the `releaseOrder()` API in the *Selling and Fulfillment Foundation: Javadocs*.

Important: Sterling Commerce 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-98 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-99 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 - Default value. Releases line quantities only after release date criteria have been met.
CheckInventory	Optional. Determine whether inventory should be checked. Valid values are: <ul style="list-style-type: none"> • Y - Default value. Inventory needs to be checked. • N - Inventory does not need to be checked.

Table A–99 Release Criteria Parameters

Parameter	Description
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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–100 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 (`<=`) 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 under the `releaseOrder()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.30 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–101 *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–102 *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.
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.

Table A–102 Route Shipment Criteria Parameters

Parameter	Description
ColonyID	Required in a multischema deployment where YFS_SHIPMENT table may exist in multiple schemas. Runs the agent for the colony.
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–103 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–104 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.3.31 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: Sterling Commerce 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–105 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–106 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.

Table A–106 Schedule Criteria Parameters

Parameter	Description
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 - Default value. Only schedules line quantities. <p>Note: Enabling this parameter does not validate hold types configured for the release transaction.</p>
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	Not used. This agent updates a failed task so that it is suspended for the back order retry interval setup in the appropriately scheduled rule.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-107 Schedule Statistics

Statistic Name	Description
NumFutureDateFailures	<p>Number of orders that Selling and Fulfillment Foundation 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-107 Schedule Statistics

Statistic Name	Description
NumOrdersCannotBeProcessedFailures	<p>Number of orders that Selling and Fulfillment Foundation 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	<p>Number of orders that have at least one line that was scheduled.</p> <p>Note: This includes scheduled lines in any status except BACKORDER.</p>
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 (`<=`) 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 under the `scheduleOrder()` API in the *Selling and Fulfillment Foundation: Javadocs*.

Providing Oracle Hints

You can provide Oracle Hints to increase the performance of the `scheduleOrder` agent. The two hints that can be provided for each criteria ID of the `scheduleOrder` agent are the Outer Hint and the Inner Hint. The Outer Hint is always used for the `YFS_TASK_Q` table. The Inner Hint is used for the `YFS_ORDER_HEADER` table only if the earlier hold functionality is used; otherwise, the Inner Hint is used for the `YFS_ORDER_RELEASE_STATUS` table.

Insert the following entries in the `yfs.properties` file in order to enable Oracle Hints:

1. Edit the `<YANTRA_HOME>/Applications/Foundation/resources/yfs.properties` file.
2. Insert `yfs.<agent_criteria_id>.getjobs.hint.outer=/*+ parallel(YFS_TASK_Q 8) full(yfs_task_q) */`

Insert `yfs.<agent_criteria_id>.getjobs.hint.inner=/*+ NL_SJ */`

A.3.32 Send Invoice

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

In environments that require an interface with accounts receivable systems, this transaction needs to be scheduled. This transaction 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 takes care of payment collection. Invoices can be published as soon as they are created.
- Publish invoice after payment collection - This implies that the Console take care of the payment collection. When payment is in the AT_COLLECT status and the payment is not from an external system, an invoice is published only if the entire payment amount is collected. If the payment is in the AT_CREATE status or the payment is from an external system, the invoice is published unconditionally.

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-108 *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–109 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–110 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–111 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 details provided under this API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.33 Send Item Changes

In integrated environments, this transaction publishes item data changes that are directed to an external system.

When item changes occur in Selling and Fulfillment Foundation, they need to be communicated to the external system.

The business process may require the synchronization of items all at once in a batch. For example, at the end of each business day, the `sendItemChanges` agent can be configured to synchronize items based on the synchronization logic. This transaction retrieves all items that are not logical kit or dynamic physical kit items and whose `SyncTS` is null or `MaxModifyTS` is greater than the `SyncTS`.

Note: The `MaxModifyTS` of an item is updated with the current timestamp whenever an item is modified. The transaction then retrieves detailed information about those items and raises the `ON_SUCCESS` event. This event should be configured to invoke the Send Item Changes action.

For more information about how this integration is implemented, see the *Selling and Fulfillment Foundation: Integration Guide*.

Attributes

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

Table A-112 *Send Item Changes Attributes*

Attribute	Value
Base Transaction ID	SEND_ITEM_CHANGES
Base Document Type	None
Base Process Type	General

Table A–112 Send Item Changes Attributes

Attribute	Value
Abstract Transaction	No
APIs Called	None

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–113 Send Item Changes 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.
Organization Code	Optional. The organization from which items are synchronized. This field is blank by default.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

For this transaction the pending job count is the number of items requiring synchronization. This is determined for product items that are not logical kit or dynamic physical kit items and whose SyncTS is null or MaxModifyTS is greater than the SyncTS.

Events Raised

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

Table A–114 Events Raised by the Send Item Changes Transaction

Transaction/Event	Key Data	Data Published	Template Support?
ON_SUCCESS	None	YCM_SEND_ITEM_CHANGES_ON_SUCCESS.XML	Yes

A.3.34 Send Customer Changes

In integrated environments, this transaction publishes customer data changes that are directed to an external system.

When customer changes occur in Selling and Fulfillment Foundation, they need to be communicated to the external system.

The business process may require the synchronization of customers all at once in a batch. For example, at the end of each business day, the `sendItemChanges` agent can be configured to synchronize items based on the synchronization logic. This transaction retrieves all customers that are consumers, have a user ID present, and are required to synchronize. This transaction can also be used to complete the initial synchronization of users between the two systems. For example, if an external system is already in place, and Selling and Fulfillment Foundation is then added, the `SendCustomerChanges` agent synchronizes the users from the external system.

The `sendCustomerChanges` agent also serves as a backup mechanism. If a customer synchronization event fails, the agent automatically retries the synchronization after a specified amount of time.

Note: The `MaxModifyTS` of an customer is updated with the current timestamp whenever an customer is modified, whenever `syncTS` is less than `MaxModifyTS`, or when `syncTS` is null. The transaction then retrieves detailed information about those customers and raises the `ON_SUCCESS` event. This event should be configured to invoke the `Send Customer Changes` action.

For more information about how this integration is implemented, see the *Selling and Fulfillment Foundation: Integration Guide*.

Attributes

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

Table A–115 *Send Customer Changes Attributes*

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

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–116 *Send Customer Changes 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.
Organization Code	Optional. The organization from which customers are synchronized. This field is blank by default.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

For this transaction the pending job count is the number of customers requiring synchronization. This is determined for customers that are consumers, have a user ID present, and are required to synchronize.

Events Raised

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

Table A–117 *Events Raised by the Send Customer Changes Transaction*

Transaction/Event	Key Data	Data Published	Template Support?
SEND_CUSTOMER_CHANGES.ON_SUCCESS	None	YSC_SEND_CUSTOMER_CHANGES.ON_SUCCESS.XML	Yes

A.3.35 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 details provided under the `sendOrder()` API in the *Selling and Fulfillment Foundation: 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-118 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-119 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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

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

value less than or equal to (\leq) the current date value in the YFS_Task_Q table.

Events Raised

This transaction raises events as specified under the `sendOrder()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.36 Send Release

The Send Release Agent dispatches releases to ship nodes.

Attributes

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

Table A–120 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–121 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.

Table A–121 Send Release Criteria Parameters

Parameter	Description
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.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–122 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–123 Events Raised by the Send Release Transaction

Transaction/Event	Data Published
PUBLISH_SHIP_ADVICE	YFS_publishShipAdvice_output.xml

A.3.37 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-124 Start Order Negotiation Attributes

Attribute	Value
Base Transaction ID	START_ORD_NEGOTIATION
Base Document Type	Order
Base Process Type	Order Fulfillment
Abstract Transaction	No
APIs Called	createNegotiation()
User Exits Called	YCPBeforeCreateNegotiationUE, YCPGetNegotiationNoUE

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A-125 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.

Table A–125 Start Order Negotiation Criteria Parameters

Parameter	Description
Node	Required. The warehouse management ship node for which records are being processed.
ColonyID	Required in a multischema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–126 Start Order Negotiation Statistics

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

Pending Job Count

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

Events Raised

This transaction raises events as specified under the `createNegotiation()` API in the *Selling and Fulfillment Foundation: Javadocs*.

A.3.38 Synchronize Colony Map

The Colony Map Synchronizer agent inserts or updates colony mappings of organizations and users in the PLT_COLONY_MAP table. When you run the agent for the first time, it populates this table, which is a necessary step in upgrading to multischema mode after installing or upgrading Selling and Fulfillment Foundation.

For more information about upgrading to multischema mode, see the *Selling and Fulfillment Foundation: Multitenant Enterprise Guide*.

Attributes

The following are attributes for this time-triggered transaction:

Table A-127 Colony Map Synchronizer Attributes

Attribute	Value
Base Transaction ID	COLONY_MAP_SYNC
Base Process Type	General
Abstract Transaction	No

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A-128 Colony Map Synchronizer 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.
ColonyID	The colony to be synchronized. Initially, you must run the agent on the DEFAULT colony provided by the Selling and Fulfillment Foundation installation so that it populates the PLT_COLONY_MAP table. After this, you can run the agent on another ColonyID.
InsertDefaultMappings	If set to Y, users for which the colony cannot be determined will be mapped to the colony for which the Colony Map Synchronizer agent is run.

Statistics Tracked

None.

Pending Job Count

None.

Events Raised

None.

Tables Purged

None.

A.3.39 Update Best Match Region

The Update Best Match Region transaction manages the YFS_REGION_BEST_MATCH table, which is used by Data Warehouse Analytics to report best match region data. The best match region is defined by the following five address attributes in person info records:

- ADDRESS_LINE6
- CITY
- STATE
- SHORT_ZIP_CODE
- COUNTRY

The agent for the Update Best Match Region transaction runs in two modes that allow you to set up and update the YFS_REGION_BEST_MATCH table.

Attributes

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

Table A-129 Update Best Match Region Attributes

Attribute	Value
Base Transaction ID	UPDATE_BEST_MATCH_REGION
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YSCGetShortZipCode UE

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–130 Update Best Match Region 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 UpdateOnly = N, only distinct records are returned per agent call. If left blank, it defaults to 1000.
TableType	<p>Required in a multischema deployment when YFS_Person_Info table may exist in multiple schemas.</p> <p>Valid Values: CONFIGURATION, TRANSACTION, MASTER.</p> <p>If set to CONFIGURATION, the agent runs for the YFS_Person_Info records associated with tables that have TableType as CONFIGURATION; for example, YFS_Organization, YFS_Ship_Node, and so forth.</p> <p>If set to TRANSACTION, the agent runs for the YFS_Person_Info records associated with tables that have TableType as TRANSACTION; for example, YFS_Order_Header, YFS_Shipment, and so forth.</p> <p>Note that the agent would run for all TableTypes that exist in the same schema as the one passed. For example, if set to TRANSACTION, the agent would also run for YFS_Person_Info records associated with tables that have TableType as MASTER, since they reside in the same schema.</p>
ColonyID	Required in a multi schema deployment where the YFS_PERSON_INFO table may exist in multiple schemas. Runs the agent for the colony.

Table A–130 Update Best Match Region Criteria Parameters

Parameter	Description
UpdateOnly	<p>Mode in which to run. Valid values are:</p> <ul style="list-style-type: none"> • N - Default value. Adds records from the YFS_PERSON_INFO table to the YFS_REGION_BEST_MATCH table and populates the region key in the YFS_BEST_MATCH table. To perform the initial setup of Best Match Region for Analytics, set UpdateOnly to N. • Y - Update mode. Updates region keys based on addresses in YFS_REGION_BEST_MATCH. After performing the initial setup of Best Match Region for Analytics, set this value to Y to specify update mode.
LastPersonInfoKey	<p>Optional. If UpdateOnly is set to N, LastPersonInfoKey determines the first person info record to populate. If no key is specified, the value defaults to Null.</p>
LastRegionBestMatchKey	<p>Optional. If UpdateOnly is set to Y, LastRegionBestMatchKey determines the first region best match key to update. If no key is specified, the value defaults to Null.</p>

Statistics Tracked

None.

Pending Job Count

None.

Events Raised

None.

Tables Purged

None.

A.3.40 PopulateOwnershipTransferSummary

This method updates the YFS_OWNERSHIP_TRANSFER_SUMMARY table.

This transaction updates the YFS_OWNERSHIP_TRANSFER_SUMMARY table by checking the records in YFS_INV_OWN_TRANSFER_RCD table.

It also updates the IS_STATISTICS_UPDATED to 'Y' in YFS_INV_OWN_TRANSFER_RCD table after the record has been used by the transaction.

Attributes

Following are the attributes for this time-triggered transaction:

Table A–131 YFSPopulateOwnershipTransfer Attributes

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

Criteria Parameters

Following are the criteria parameters for this transaction:

Table A–132 YFSPopulateOwnershipTransfer Criteria Parameters

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, which is 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.
ColonyID	Required in a multi schema deployment where the YFS_OWNERSHIP_TRANSFER_SUMMARY and YFS_INV_OWN_TRANSFER_RCD tables may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None

Pending Job Count

None

Events Raised

None

A.4 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 is not accurate in the Order Shipment Console.

Note: Some of the statistics collected and tracked in Release 8.5 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Selling and Fulfillment Foundation.

Note: All Time-Triggered Purge Transactions have a `CollectPendingJobs` criteria parameter. If this parameter is set to `N`, the agent does not collect information on the pending jobs for that time-triggered transaction. This pending job information is used for monitoring the monitor in the System Management Console.

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.4.1 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.4.2 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 a purge log file directory:

1. Configure the `yfs.purge.path` property in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For example, on UNIX you might specify the log files to be written to the `/app/yfs/logs/purges` directory.

For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

2. Run the `<INSTALL_DIR>/bin/setupfiles.sh` script on UNIX, or the `<INSTALL_DIR>/bin/setupfiles.cmd` script on Windows.

A.4.3 Available Purges

This section contains details of all purge transactions in alphabetical order. The time-triggered purge transactions are:

- [Alert Purge](#)
- [Capacity Purge](#)
- [Draft Order History Purge](#)
- [Draft Order Purge](#)
- [Delivery Plan Purge](#)
- [Export Table Purge](#)
- [Import Table Purge](#)
- [Inventory Audit Purge](#)
- [Inventory Purge](#)
- [Inventory Supply Temp Purge](#)
- [Item Audit Purge](#)
- [Load History Purge](#)
- [Load Purge](#)
- [Negotiation History Purge](#)
- [Negotiation Purge](#)
- [Order History Purge](#)
- [Order Purge](#)
- [Order Release Status Purge](#)
- [Order Status Audit Purge](#)
- [Organization Audit Purge](#)
- [Person Info Purge](#)
- [Person Info History Purge](#)
- [Picklist Purge](#)
- [Price List Purge](#)
- [Purge Catalog Mass Audits](#)

- Receipt History Purge
- Receipt Purge
- Reprocess Error Purge
- Reservation Purge
- Shipment History Purge
- Shipment Purge
- Shipment Statistics Purge
- Statistics Purge
- User Activity Purge
- User Activity Audit Purge
- Work Order History Purge
- Work Order Purge
- YFS Audit Purge
- YFSInventoryOwnershipAudit Purge
- Password Reset Request Purge
- User Login Failure Purge

A.4.3.1 Alert Purge

This purge removes alert data from the system. This reduces the load on frequently accessed tables. The alert should be marked as CLOSED.

Any enterprise that uses the Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, an alert is picked up for purge:

- The alert should be in "Closed" status.
- An inactive alert should have the resolution date earlier than or equal to the current date minus the purge criteria's retention days.
- If the alert is in "Open" status, the number of expiration days should be greater than 0, and the modified timestamp should be less than the current date minus the number of expiration days.

Attributes

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

Table A–133 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–134 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 does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
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.
ExceptionsWithBlank EnterpriseOnly	Optional. If the parameter is set to Y, the agent purges only those exceptions that has blank enterprise code. In this case, the value set for the EnterpriseCode criteria parameter is ignored.

Table A–134 Alert Console Purge Criteria Parameters

Criteria Parameters	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 are 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–135 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_NOTES

YFS_INBOX_AUDIT

YFS_INBOX_REFERENCES

A.4.3.2 Capacity Purge

This purge removes capacity data from the system. This reduces the load on frequently accessed tables.

Any enterprise using the Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, a capacity data gets picked up for purge:

- All resource pool standard capacity periods with effective to date earlier than or equal to the current date minus the purge criteria's retention days.
- All resource pool overridden capacity with the capacity date earlier than or equal to the 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 resource pool capacity consumption details where appointment date is earlier than the system date minus the purge criteria's retention days (or ManualReservationPurgeLeadDays for manually created reservations).
- All resource pool capacity consumption details where expiration date has passed and reservation Id is not blank.

Attributes

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

Table A–136 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–137 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 are moved to history tables without actually moving them.

Table A–137 Capacity 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–138 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 Retention Days). When this table is purged, YFS_RES_POOL_CONSMPTN_DTLS is also purged.

The YFS_RES_POOL_CONSMPTN_DTLS table is purged when RESERVATION_EXPIRATION_DATE \leq (CurrentDate - LeadDays)

A.4.3.3 Draft Order History Purge

This purge deletes data from history tables after a specified interval, which in turn, reduces the load on frequently accessed tables.

You can use purge codes' pseudo-logic to analyze the purges. If the following condition is met, a draft order is picked up for history purge:

- The last modified date of the draft order exceeds the retention day period.

All the enterprise using the Console must schedule purge transactions.

For more information about Additional Purge Criteria Based on Line Type, see the *Sterling Distributed Order Management: Configuration Guide*.

Note: The draft order must be purged and moved to the history tables before you purge the draft order history tables. See [Section A.4.3.4, "Draft Order Purge"](#).

Note: Selling and Fulfillment Foundation does not provide a transaction for draft order history purges. If you are defining a transaction that purges draft order history tables, refer to ["Criteria Parameters"](#) on page 610 for information about the transaction criteria.

If you do not want to define your own transaction to purge draft order history tables, you can use the Order Purge transaction and specify DRAFTORDERHISTPRG for the PurgeCode. To configure the Order Purge transaction for draft order history table purges, refer to ["Order Purge"](#) on page 648 for more information.

Criteria Parameters

The following are the criteria parameters for defining a draft order history transaction:

Table A–139 Draft 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.
EnterpriseCode	Required. Enterprise for which the Draft Order History Purge has to be run. If not passed, all the 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 that are listed in Tables Purged. N - Test mode. Determines the rows that are removed without actually removing them.
PurgeCode	Required. Set to DRAFTORDERHISTPRG. Used for internal calculations, such as determining retention days. Corresponds to the PurgeCode used in Business Rules Purge Criteria.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Events Raised

None.

Tables Purged

YFS_ANSWER_SET_TRAN_H

YFS_ANSWER_TRAN_H
YFS_CHARGE_TRAN_DIST_H
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_PROMOTION_AWARD_H
YFS_PROMOTION_H
YFS_RECEIVING_DISCREPANCY_DTL_H
YFS_RECEIVING_DISCREPANCY_H
YFS_REFERENCE_TABLE_H
YFS_TAX_BREAKUP_H

A.4.3.4 Draft Order Purge

This purge archives data into history tables after a specified interval, which in turn, reduces the load on frequently accessed tables. For information about purging draft orders from history tables, see [Section A.4.3.3, "Draft Order History Purge"](#).

NOTE: Selling and Fulfillment Foundation does not provide a transaction for draft order purges. If you are defining a transaction that purges draft orders, refer to [Criteria Parameters on page 614](#) for details about the transaction criteria.

If you do not want to define your own transaction to purge draft orders, you can use the Order Purge transaction and specify DRAFTORDERPRG for the PurgeCode. To configure the Order Purge transaction for draft order purges, refer to ["Order Purge"](#) on page 648 for more information.

All the enterprise using the Console must schedule purge transactions.

Draft orders are picked up by the agent for validation when the following conditions are met:

- Draft order flag is set to Y.
- Modifyts is set for the retention date.

After the draft orders are picked up, each draft order is validated for purging based on the following conditions:

- No eligible order release status records (records with a status larger than zero) exist for the order.
- All the open child orders (derived, chained, return, exchange, or refund fulfillment) for the order are already purged.

If a draft order meets the set of conditions for validation listed earlier, the agent continues to verify the draft orders against the following criteria:

- Contains the Draft Created (1000) status, and all the extended Draft Created statuses.
- Does not have an order release status record that does not meet the retention days.
- The order's last modification should be before the lead time (in days) setup.
- In the case when an exchange order is part of a return order, the exchange order should be purged from history tables before the return order is purged.
- In the case of an order line reservation, the draft order cannot be purged.
- If the Draft Order Payment Processing flag is set to N, the draft orders are purged.
- If the Draft Order Payment Processing flag is set to Y and a charge exists on a draft order, the draft order is not purged. However, authorizations are not considered when validating draft orders for purge.
- For order lines, except service order 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 check box enabled.)
 - If the Seller Demand Type is blank.
 - If the Buyer inventory update is required, and the Buyer Supply Type is Onhand, or blank.

Criteria Parameters

The following are the criteria parameters for defining a draft order purge transaction:

Table A–140 Draft 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	Required. Enterprise for which the Draft Order Purge has to be run. If not passed, all the enterprises are monitored. Note: When the EnterpriseCode is blank, the purge criteria configured for the DEFAULT enterprise is used, and not the purge criteria configured for the draft order's enterprise.
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 are moved to history tables without actually moving them.
PurgeCode	Required. Set to DRAFTORDERPRG. Used for internal calculations, such as determining retention days. Corresponds to the PurgeCode used in Business Rules Purge Criteria.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Events Raised

None.

Tables Purged

YFS_ACTIVITY_DEMAND
YFS_ANSWER_SET_TRAN
YFS_ANSWER_TRAN
YFS_CHARGE_TRANSACTION
YFS_CHARGE_TRAN_DIST
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_RESERVATION
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_PMNT_TRANS_ERROR
YFS_PROMOTION
YFS_PROMOTION_AWARD
YFS_RECEIVING_DISCREPANCY
YFS_RECEIVING_DISCREPANCY_DTL
YFS_REFERENCE_TABLE
YFS_TAX_BREAKUP

A.4.3.5 Delivery Plan Purge

This purge deletes delivery plans after they have completed their typical life-cycle. 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 (in days) setup.

Any enterprise using the Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, a delivery plan is picked up for purge:

- The delivery plan should be in the "Closed" status.
- The delivery plan should not be associated with any load or shipment.
- All orders associated with the delivery plan should be purged.
- The last modification performed on the delivery plan should fall before the lead time (in days) setup.

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–141 *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–142 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.
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. 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 are 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.
BatchDelete	Required. The method by which all records are deleted from the table. Valid values are: <ul style="list-style-type: none"> Y - Default value. Records are deleted in batches. N - Records are deleted one by one.
ColonyID	Required in a multi schema deployment where the YFS_DELIVERY_PLAN table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–143 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.4.3.6 Export Table Purge

This purge removes export table data from the system. This reduces the load on frequently accessed tables.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, the YFS_EXPORT table is picked up for purge:

- YFS_EXPORT records should be marked as processed (Status = 10).
- The last modified time should fall before the lead time (in days) setup.

Note: This purge only reads the rules defined by the hub. Enterprise overridden rules are not considered. This purge should be single threaded when you run it in batch delete mode(BatchDelete=Y).

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–144 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–145 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 are moved to history tables without actually moving them.
BatchDelete	Required. The method by which all records are deleted from the table. Valid values are: <ul style="list-style-type: none"> Y - Records are deleted in batches. N - Default value. Records are deleted one by one.

Table A–145 Export Table 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.
CollectPendingJobs	If this parameter is set to "N", the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
ColonyID	Required in a multi schema deployment where the YFS_EXPORT table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–146 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.4.3.7 Import Table Purge

This purge removes import table data from the system. This reduces the load on frequently accessed tables.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, the YFS_IMPORT table is picked up for purge:

- YFS_IMPORT records should be marked as processed (Status = "10").
- The "last modified time" should fall before the lead time (in days) setup.

Note: This purge only reads the rules defined by the hub. Enterprise overridden rules are not considered. This purge should be single threaded when you run it in batch delete mode(BatchDelete=Y).

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A-147 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–148 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 are moved to history tables without actually moving them.
BatchDelete	Required. The method by which all records are deleted from the table. Valid values are: <ul style="list-style-type: none"> • Y - Records are deleted in batches. • N - Default value. Records are deleted one by one.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
CollectPendingJobs	If this parameter is set to "N", the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
ColonyID	Required in a multi schema deployment where the YFS_IMPORT table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-149 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.4.3.8 Inventory Audit Purge

This purge removes inventory audit data from the system. This reduces the load on frequently accessed tables.

Any enterprise using the Console must schedule purge transactions.

All inventory audits of the provided organization with modify timestamp earlier than the current date minus the purge criteria's retention days can be configured to be picked up by the Inventory Audit Purge.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, an inventory audit record is picked up for purge:

- The inventory audit record's last modification is earlier than the current timestamp minus the retention days.

Note: Number of threads for this purge's agent criteria details must be set to 1. For more information about agent criteria, see the appropriate section in this guide.

Important: The Inventory Audit purge does 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–150 *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–151 *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.

Table A–151 Inventory Audit 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 Table Purged to the corresponding history tables. N - Test mode. Determines the rows that are 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–152 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.4.3.9 Inventory Purge

This purge removes inventory data from the system. This reduces the load on frequently accessed tables. This purge does not take retention days into account when purging.

You can use purge codes pseudo-logic to analyze purges.

For YFS_INVENTORY_SUPPLY, if the following conditions are met, an inventory supply is picked up for purge:

- Supply record has the same availability type as the node. For example, TRACK or INFINITE.
- Supply record has 0 quantity.
- Supply record does not contain the supply type "INFO".

For YFS_INVENTORY_DEMAND, if the following conditions are met, an inventory demand is picked up for purge:

- Demand record has 0 quantity or lesser.
- Demand record does not have demand details as well as matching demand record in YFS_INVENTORY_DEMAND_ADDNL tables.

For YFS_INVENTORY_TAG, it is purged if the INVENTORY_TAG_KEY is not used by any of the existing supply and demand.

For YFS_INVENTORY_RESERVATION, an inventory reservation is picked up for purge if it meets the following conditions:

- Inventory reservation record has 0 quantity or ship date is earlier than the system date minus the purge criteria's retention days.

For YFS_INVENTORY_NODE_CONTROL, it is purged if the INV_PIC_INCORRECT_TILL_DATE is earlier than the current time stamp minus the purge criteria's retention days.

For YFS_IBA_TRIGGER, it is purged if IBA_REQUIRED = 'N', IBA_RUN_REQUIRED = 'N', and LAST_IBA_PROCESSED_TS is earlier than the current time stamp minus the purge criteria's retention days.

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–153 Inventory Purge Attributes

Attribute	Value
Base Transaction ID	INVENTORYPRG
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–154 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 are moved to history tables without actually moving them.

Table A–154 Inventory 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–155 Inventory Purge Statistics

Statistic Name	Description
NumInventoryDemandsPurged	Number of inventory demands purged.
NumInventoryNodeControlsPurged	Number of inventory node controls purged.
NumInventoryReservationsPurged	Number of inventory reservations purged.
NumInventoryTagsPurged	Number of inventory tags purged.
NumItemBasedAllocationTriggersPurged	Number of item based allocation triggers 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, YFS_Inventory_Reservation, YFS_IBA_Trigger, and YFS_Inventory_Node_Control tables.

Events Raised

None.

Tables Purged

YFS_IBA_TRIGGER

YFS_INVENTORY_DEMAND

YFS_INVENTORY_TAG

YFS_INVENTORY_RESERVATION

YFS_INVENTORY_SUPPLY

YFS_INVENTORY_NODE_CONTROL

A.4.3.10 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 the Selling and Fulfillment Foundation 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 about configuring the synchronization with node inventory, see the *Sterling Global Inventory Visibility: Configuration Guide*.

The Inventory Supply Temp purge agent is used to purge all records in the YFS_INVENTORY_SUPPLY_TEMP table whose modify timestamp is less than current time minus the purge criteria's retention days for a group of YantraMessageGroupID.

Attributes

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

Table A-156 Inventory Supply Temp Purge Attributes

Attribute	Value
Base Transaction ID	SUPPLYTEMPPRG
Base Document Type	General
Base Process Type	General

Table A–156 Inventory Supply Temp Purge Attributes

Attribute	Value
Abstract Transaction	No
APIs Called	None
User Exits Called	None

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–157 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.
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 are 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.
ColonyID	Required in a multi schema deployment where the YFS_INVENTORY_SUPPLY_TEMP table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–158 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.4.3.11 Item Audit Purge

This purge removes the YFS_AUDIT table data from the system, which reduces the load on frequently accessed tables. It purges records in the YFS_AUDIT and the YFS_AUDIT_HEADER tables that meet the following conditions:

- YFS_AUDIT records that have 'modifyts' greater than the retention days specified and the records have the table name as 'YFS_ITEM'.
- The last modified time is before the lead time (in days) setup.

When the enterprise modifies records in the YFS_ITEM table through the Applications Manager, the YFS_ITEM is audited and the audit records are inserted in the YFS_AUDIT table. In order to clean up the audit records, this purge transaction can be used.

Any enterprise using the Console must schedule purge transactions accordingly.

Attributes

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

Table A–159 *Item Audit Purge Attributes*

Attribute	Value
Base Transaction ID	YFS_ITEM_AUDIT_PURGE
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–160 *Item Audit Purge Criteria Parameters*

Parameter	Description
Action	Required. Triggers the transaction. If left blank, the value 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), this value defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> Y - Default value. Production mode. Deletes records from the regular tables. N - Test mode.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
ColonyID	Required in a multi schema deployment where the YFS_AUDIT and YFS_AUDIT_HEADER tables may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-161 Item Audit Purge Statistics

Statistic Name	Description
NumItemAuditRecordsPurged	Number of item audit records purged.

Pending Job Count

For this transaction, the pending job count is the number of records that can be purged from the YFS_AUDIT table that match the criteria values.

Events Raised

None.

Tables Purged

YFS_AUDIT, YFS_AUDIT_HEADER

A.4.3.12 Load History Purge

This purge deletes the load data from history tables after it completes its typical lifecycle. This reduces the load on frequently accessed tables.

Any enterprise using the Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, a load is picked up for purge:

- The last modification made to the load is before the lead time (in days) setup.

Note: Before you run this transaction, ensure to purge loads and move them to history tables. For more information about purging loads, see [Section A.4.3.13, "Load Purge"](#).

Attributes

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

Table A–162 Load History Purge Attributes

Attribute	Value
Base Transaction ID	LOADHISTPRG
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–163 Load 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 Load Purge needs to be run. If not passed, 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 are moved to history tables without actually moving them.

Table A–163 Load History Purge Criteria Parameters

Parameter	Description
Purge Code	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–164 Load History Purge Statistics

Statistic Name	Description
NumLoadHistoriesPurged	Number of load histories purged.
NumLoadShipmentHistoriesPurged	Number of load shipment histories purged.

Pending Job Count

For this transaction, the pending job count is the number of records that can be purged from the YFS_Load_H table.

Events Raised

None.

Tables Purged

YFS_LOAD_H

YFS_LOAD_STOP_H

YFS_LOAD_SHIPMENT_CHARGE_H

YFS_LOAD_STATUS_AUDIT_H

YFS_SHIPMENT_CONTAINER_H

YFS_CONTAINER_ACTIVITY_H

YFS_LOADED_CONTAINER_H
YFS_LOAD_SHIPMENT_H
YFS_ADDITIONAL_DATE_H
YFS_LOAD_HOLD_TYPE_H
YFS_LOAD_HOLD_TYPE_LOG_H

A.4.3.13 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 Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, a load is picked up for purge:

- The Load's last modification should fall before the lead time (in days) setup.

Attributes

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

Table A–165 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–166 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.
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 are moved to history tables without actually moving them.

Table A-166 Load 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-167 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_HOLD_TYPE

YFS_LOAD_HOLD_TYPE_LOG

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.4.3.14 Negotiation History Purge

This purge deletes negotiation history data from the system. This reduces the load on frequently accessed tables. It purges data from the order negotiation history tables.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, a negotiation is picked up for history purge:

- The last modified date of the negotiation exceeds the retention day period.

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A-168 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–169 *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 are 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–170 *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_AUDIT

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.4.3.15 Negotiation Purge

This purge archives data into history tables after it completes its typical lifecycle. This reduces the load on frequently accessed tables. It works from the task queue (YFS_TASK_Q) table.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, a negotiation is picked up for purge:

- The last modification performed on the negotiation falls before the lead time (in days) setup.
- The negotiation is in pickable status.

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–171 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–172 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 are moved to history tables without actually moving them.

Table A–172 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–173 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_AUDIT

YFS_NEGOTIATION_HDR

YFS_NEGOTIATION_LINE

YFS_RESPONSE

YFS_RESPONSE_HDR
 YFS_RESPONSE_LINE
 YFS_RESPONSE_LINE_DTL

A.4.3.16 Order History Purge

This purge deletes data from history tables after it completes its typical lifecycle. This reduces the load on frequently accessed tables.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, an order is picked up for history purge:

- The last modified date of the order exceeds the retention day period.

Any enterprise using the Console must schedule purge transactions.

For more information about Additional Purge Criteria Based on Line Type, see the *Sterling Distributed Order Management: Configuration Guide*.

Note: The order should have been purged and moved into the history tables before you can run this transaction. See [Section A.4.3.17, "Order Purge"](#).

Attributes

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

Table A–174 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–175 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.
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 are 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–176 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_ANSWER_SET_TRAN_H
YFS_ANSWER_TRAN_H
YFS_CHARGE_TRAN_DIST_H
YFS_CHARGE_TRAN_REQUEST_H
YFS_CHARGE_TRAN_RO_MAP_H
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_PROMOTION_AWARD_H
YFS_PROMOTION_H
YFS_RECEIVING_DISCREPANCY_DTL_H
YFS_RECEIVING_DISCREPANCY_H
YFS_REFERENCE_TABLE_H
YFS_TAX_BREAKUP_H

A.4.3.17 Order Purge

This purge archives data into history tables after it completes its typical lifecycle. To purge orders from history tables, see [Section A.4.3.16, "Order History Purge"](#). This reduces the load on 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.

Note: If purge criteria are not met, AVAILABLE_DATE is calculated based on the modify time stamp of the order in YFS_ORDER_HEADER table as well as the YFS_TASK_Q table, whichever is maximum. To this value, retention days is added to the new AVAILABLE_DATE.

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.
- Released (3200) and all extended Released statuses.
- Shipped (3700) and all extended Shipped statuses.
- Completed (3700) and all extended Completed statuses. These statuses are available only for the document type Master Order.
- Received (3900) and all extended Received statuses.
- Cancelled (9000) and all extended Cancelled statuses.
- Shorted (9020) and all extended Shorted statuses.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, an order is picked up for purge:

- All open child orders (derived, chained, return, exchange, procurement, or refund fulfillment) for the order must already be purged.
- No pending transfer-out charges to another order exceeding the transfer-ins.
- No pending adjustment invoices.

An order is purged immediately if it meets the above three criteria and is completely cancelled with payment collection complete.

Note: In order for the purge agent to pick up a cancelled order, the payment status of the order must be one of the following:

- Paid
- Not Applicable

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 retention days.
- It should be in the correct status for purge. For example,
 - All service requests for the order should have Shipped or extended Shipped status.
 - The payment status for the order should be Paid 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 (in days) setup.
- Any enterprise using the Console 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.

When the purge agent is run, the draft order without lines are purged to the order history table. Once the purge history agent is run, the draft orders without lines gets deleted permanently.

Attributes

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

Table A-177 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-178 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. Note: When the EnterpriseCode is blank, the purge criteria configured for the DEFAULT enterprise is used; not the purge criteria configured for the order's enterprise.
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 are moved to history tables without actually moving them.

Table A–178 Order Purge Criteria Parameters

Parameter	Description
PurgeCode	<p>Required. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria. You can set this parameter to the following values:</p> <ul style="list-style-type: none"> • DRAFTORDERHISTPRG to purge draft order information from the order history tables. • DRAFTORDERNOLINEHISTPRG to purge draft orders without order lines from the order history tables. • DRAFTORDERNOLINEPRG to purge draft orders that have no order lines. • DRAFTORDERPRG to purge draft order information and archive it in the order history tables. <p>PurgeCode cannot be set to the value ORDER_RELEASE_STATUS_PURGE.</p>
AdditionalPurgeCode	<p>Optional. To purge order release status records, set this parameter to ORDER_RELEASE_STATUS_PURGE.</p> <p>For more information, see Section A.4.3.18, "Order Release Status Purge".</p>
ColonyID	<p>Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.</p>

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–179 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_ACTIVITY_DEMAND
YFS_ANSWER_SET_TRAN
YFS_ANSWER_TRAN
YFS_CHARGE_TRANSACTION
YFS_CHARGE_TRAN_DIST
YFS_CHARGE_TRAN_REQUEST
YFS_CHARGE_TRAN_RQ_MAP
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_RESERVATION
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_PMNT_TRANS_ERROR
YFS_PROMOTION
YFS_PROMOTION_AWARD
YFS_RECEIVING_DISCREPANCY
YFS_RECEIVING_DISCREPANCY_DTL
YFS_REFERENCE_TABLE
YFS_TAX_BREAKUP

A.4.3.18 Order Release Status Purge

The Order Release Status Purge agent extends the Order Purge agent's capabilities by purging order release status records before the Order Purge agent completely purges data to history tables.

If an order meets the criteria for purging, the order release status records with quantities of 0 are deleted from the YFS_ORDER_RELEASE_STATUS table and are not put into the history table.

When the Order Release Status Purge agent has completed, the task queue's AVAILABLE_DATE is reset to the date specified by the purge criteria for Order Purge. This enables the Order Purge agent to pick up and process an order as necessary. Order Purge will continue to purge order release status records as usual.

If the following conditions are met, the Order Purge agent purges order release status records:

- All conditions for Order Purge have been met. See [Section A.4.3.17, "Order Purge"](#) for information about conditions for Order Purge.
- Order release records have 0 quantity.
- AdditionalPurgeCode in the Order Purge criteria is set to ORDER_RELEASE_STATUS_PURGE.
- The order has been modified within the Order Purge lead days AdditionalPurgeCode.

Criteria Parameters

The following are the criteria parameters for Order Release Status Purge:

Table A–180 Order Release Status 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.

Table A–180 Order Release Status Purge Criteria Parameters

Parameter	Description
EnterpriseCode	Optional. Enterprise for which the Order Purge needs to be run. If not passed, then all enterprises are monitored. Note: When the EnterpriseCode is blank, the purge criteria configured for the DEFAULT enterprise is used; not the purge criteria configured for the order's enterprise.
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 are moved to history tables without actually moving them.
PurgeCode	Required. To extend the Order Purge agent to purge order release status records, set to ORDERPRG. Used for internal calculations, such as determining retention days. You must also set AdditionalPurgeCode.
AdditionalPurgeCode	Required. To purge order release status records, set this parameter to ORDER_RELEASE_STATUS_PURGE.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

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

Events Raised

None.

Tables Purged

YFS_ORDER_RELEASE_STATUS

A.4.3.19 Order Status Audit Purge

This purge removes order status audit data from the system. This reduces the load on frequently accessed tables.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, an order status audit is picked up for history purge:

- The last modified time falls before the lead time (in days) setup.

Any enterprise using the Console 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–181 Order Status Audit Purge Attributes

Attribute	Value
Base Transaction ID	STATUSAUDITPRG
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–182 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 are 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.
ColonyID	Required in a multi schema deployment where the YFS_STATUS_AUDIT Table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Pending Job Count

Table A-183 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.4.3.20 Organization Audit Purge

This purge removes the YFS_AUDIT table data from the system, which reduces the load on frequently accessed tables. It purges records in the YFS_AUDIT and the YFS_AUDIT_HEADER tables that meet the following conditions:

- The YFS_AUDIT records that have 'modifyts' greater than the retention days specified and the records have the table name as 'YFS_ORGANIZATION'.
- The last modified time is before the lead time (in days) setup.

When the enterprise modifies records in the YFS_ORGANIZATION table through the Applications Manager, the YFS_ORGANIZATION is audited and the audit records are inserted in the YFS_AUDIT table. In order to clean up the audit records, this purge transaction can be used.

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–184 Organization Audit Purge Attributes

Attribute	Value
Base Transaction ID	YFS_ORGANIZATION_AUDIT_PURGE
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–185 Organization Audit Purge Criteria Parameters

Parameter	Description
Action	Required. Triggers the transaction. If left blank, the value 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), this value defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> Y - Default value. Production mode. Deletes records from the regular tables. N - Test mode.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds to the PurgeCode used in Business Rules Purge Criteria.
ColonyID	Required in a multi schema deployment where the YFS_AUDIT and YFS_AUDIT_HEADER tables may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–186 Organization Audit Purge Statistics

Statistic Name	Description
NumOrganizationAuditRecordsPurged	Number of organization audit records purged.

Pending Job Count

For this transaction, the pending job count is the number of records that can be purged from the YFS_AUDIT table that match the criteria values.

Events Raised

None.

Tables Purged

YFS_AUDIT

YFS_AUDIT_HEADER

A.4.3.21 Person Info Purge

This purge gets a list of dates with the person info record count and sorts them by date in ascending order. Then, based on the specified number of records to buffer and the modify timestamp, it purges the applicable records and places them in the YFS_PERSON_INFO_H table.

Attributes

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

Table A–187 PersonInfo Purge Attributes

Attribute	Value
Base Transaction ID	PERSONINFOPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No

Table A–187 PersonInfo Purge Attributes

Attribute	Value
APIs Called	None
User Exits Called	None

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–188 PersonInfo 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. <ul style="list-style-type: none"> • If left blank or the number specified is less than 10000, it defaults to 5000. • If the number specified is greater than 10000, then that value is used.
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 are 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.
CollectPendingJobs	If this parameter is set to "N", the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.

Table A–188 PersonInfo Purge Criteria Parameters

Parameter	Description
EnterpriseCode	Optional. Enterprise for which the PersonInfo Purge needs to be run. If not passed, then all enterprises are monitored.
TableType	<p>Required in a multi schema deployment when YFS_Person_Info table may exist in multiple schemas.</p> <p>Valid Values: CONFIGURATION, TRANSACTION, MASTER.</p> <p>If set to CONFIGURATION, purge runs for the YFS_Person_Info records associated with tables that have TableType as CONFIGURATION; for example, YFS_Organization, YFS_Ship_Node, and so forth.</p> <p>If set to TRANSACTION, purge runs for the YFS_Person_Info records associated with tables that have TableType as TRANSACTION; for example, YFS_Order_Header, YFS_Shipment, and so forth.</p> <p>Note that purge would run for all TableTypes that exist in the same schema as the one passed. For example, if set to TRANSACTION, purge would also run for YFS_Person_Info records associated with tables that have TableType as MASTER, since they reside in the same schema.</p>
ColonyID	Required in a multi schema deployment where the YFS_PERSON_INFO table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

If it is left blank or any number less than 10,000 is specified, then it defaults to 10,000. But if any number > 10,000 is specified, then that value would be used.

Table A–189 PersonInfo Purge Statistics

Statistic Name	Description
NumPersonInfoPurged	Number of person info records purged.

Pending Job Count

For this transaction, the pending job count is the number of records that can be purged from the YFS_PERSON_INFO table.

Events Raised

None.

Tables Purged

YFS_PERSON_INFO

A.4.3.22 Person Info History Purge

This purge deletes records from the YFS_PERSON_INFO_H table based on the purge criteria.

Attributes

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

Table A–190 PersonInfo Purge Attributes

Attribute	Value
Base Transaction ID	PERSONINFOHISTPRG
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–191 PersonInfo 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. <ul style="list-style-type: none"> If left blank or the number specified is less than 10000, it defaults to 5000. If the number specified is greater than 10000, then that value is used.
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 are 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.
CollectPendingJobs	If this parameter is set to "N", the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
EnterpriseCode	Optional. Enterprise for which the PersonInfo Purge needs to be run. If not passed, then all enterprises are monitored.

Table A–191 PersonInfo Purge Criteria Parameters

Parameter	Description
TableType	<p>Required in a multi schema deployment when YFS_Person_Info table may exist in multiple schemas.</p> <p>Valid Values: CONFIGURATION, TRANSACTION, MASTER.</p> <p>If set to CONFIGURATION, purge runs for the YFS_Person_Info records associated with tables that have TableType as CONFIGURATION; for example, YFS_Organization, YFS_Ship_Node, and so forth.</p> <p>If set to TRANSACTION, purge runs for the YFS_Person_Info records associated with tables that have TableType as TRANSACTION; for example, YFS_Order_Header, YFS_Shipment, and so forth.</p> <p>Note that purge would run for all TableTypes that exist in the same schema as the one passed. For example, if set to TRANSACTION, purge would also run for YFS_Person_Info records associated with tables that have TableType as MASTER, since they reside in the same schema.</p>
ColonyID	Required in a multi schema deployment where the YFS_PERSON_INFO_H table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–192 PersonInfo Purge Statistics

Statistic Name	Description
NumPersonInfoHistoryRecordsPurged	Number of person info history records purged.

Pending Job Count

For this transaction, the pending job count is the number of records that can be purged from the YFS_PERSON_INFO_H table.

Events Raised

None.

Tables Purged

YFS_PERSON_INFO_H

A.4.3.23 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 Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, a picklist is picked up for purge:

- The picklist should exist for more than the specified retention period.
- The picklist should not be associated with any shipment.

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–193 *Picklist Purge Attributes*

Attribute	Value
Base Transaction ID	PICKLISTPRG
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–194 *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 are 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.
ColonyID	Required in a multi schema deployment where the YFS_PICK_LIST table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–195 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.4.3.24 Price List Purge

This purge removes price list data from the system. This reduces the load on frequently accessed tables.

Any enterprise using the Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, a price list is picked up for purge:

- The price list has valid date less than the current date minus the purge criteria's retention days.

Attributes

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

Table A–196 Price List Purge Attributes

Attribute	Value
Base Transaction ID	PRICELISTPRG
Base Document Type	General
Base Process Type	General

Table A–196 Price List Purge Attributes

Attribute	Value
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–197 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 are 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–198 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.4.3.25 Purge Catalog Mass Audits

This purge removes old audit records from the YFS_CATALOG_MASS_AUDIT table. This table contains data about changes to the catalog due to assignment of attributes and attribute values to categories and items. It also contains information about inherited attributes and attribute values. The purge transaction finds mass audit records that have not been modified in a specified number of days and removes those records from the database.

Attributes

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

Table A–199 Purge Catalog Mass Audits Attributes

Attribute	Value
Base Transaction ID	CATALOG_MASS_AUDIT_PURGE
Base Document Type	General

Table A–199 Purge Catalog Mass Audits 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–200 Purge Catalog Mass Audits 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 are 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.
ColonyID	Required in a multi schema deployment where the YFS_CATALOG_MASS_AUDIT table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–201 Purge Catalog Mass Audits Statistics

Statistic Name	Description
NumCatalogMassAuditsPurged	Number of mass audit records purged.

Pending Job Count

For this transaction the pending job count is the total number of records that can be purged from the YFS_CATALOG_MASS_AUDIT table.

Events Raised

None.

Tables Purged

The YFS_CATALOG_MASS_AUDIT table is purged when $\text{MODIFYTS} < (\text{CurrentDate} - \text{LeadDays})$

A.4.3.26 Receipt History Purge

This transaction deletes receipts previously archived by the Receipt Purge. See [Section A.4.3.27, "Receipt Purge"](#).

Any enterprise using the Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, a receipt that is previously purged by the receipt purge agent is picked up for history purge:

- The last modified date of the receipt should exceed the retention day period.
- The shipment associated with the receipt should be purged from the history table.

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–202 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–203 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 are 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–204 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.4.3.27 Receipt Purge

This purge removes receipt data from the system. This reduces the load on 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.4.3.26, "Receipt History Purge"](#). 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 Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, a receipt is picked up for purge:

- The last modified date of the receipt should exceed the retention day period.

- The shipment associated with the receipt should be purged.
- The receipt should be in pickable status for the purge transaction.
- The value of the OpenReceiptFlag field should be set to "N".
- The receipt should not have pending inspections.
- There is no inventory in the warehouse for the receipt.

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–205 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–206 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 are moved to history tables without actually moving them.

Table A–206 Receipt 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–207 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.4.3.28 Reprocess Error Purge

This purge deletes reprocess errors from the system. This reduces the load on frequently accessed tables.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, a YFS_REPROCESS_ERROR table is picked up for purge:

- YFS_REPROCESS_ERROR records with State = Fixed or Ignored is processed.
- The last modified time is earlier than the lead time (in days) setup.

Note: This purge only reads the rules defined by the hub. Enterprise overridden rules are not considered.

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–208 *Reprocess Error Purge Attributes*

Attribute	Value
Base Transaction ID	REPROCESSPRG
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–209 *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 are 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.
ColonyID	Required in a multi schema deployment where the YFS_REPROCESS_ERROR table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–210 *Reprocess Error Purge Statistics*

Statistic Name	Description
NumReprocessErrorsPurged	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.4.3.29 Reservation Purge

This purge deletes expired inventory reservations from the system. This reduces the load on frequently accessed tables as well as free up demands that are consumed by expired reservations.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, all records in the YFS_INVENTORY_RESERVATION tables are picked up for purge:

- EXPIRATION_DATE is earlier than the current date or quantity is less than or equal to 0

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–211 Reservation Purge Attributes

Attribute	Value
Base Transaction ID	RESERVATIONPRG
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–212 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 are 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.
ColonyID	Required in a multi schema deployment where the YFS_INVENTORY_RESERVATION table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–213 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.4.3.30 Shipment History Purge

This transaction deletes shipments previously archived by the Shipment Purge. See [Section A.4.3.31, "Shipment Purge"](#).

Any enterprise using the Console must schedule purge transactions.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, all records archived in the history table are picked up for purge:

- The last modification performed on the shipment falls before the lead time (in days) setup.

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.3.10, "Close Shipment"](#).

Attributes

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

Table A–214 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–215 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 are 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–216 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_AUDIT

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.4.3.31 Shipment Purge

This purge removes shipment data from the system. This reduces the load on 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.4.3.30, "Shipment History Purge"](#). 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.

Any enterprise using the Console 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.3.10, "Close Shipment"](#).

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, a shipment is picked up for purge:

- The last modification performed on the shipment should fall before the lead time (in days) setup.
- The value of the ShipmentClosedFlag field should be set to "Y".
- The order record should already be purged for all shipment lines.

Attributes

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

Table A-217 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–218 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.
Number of Days To Execute	Optional. Maximum number of days before the lead days the agent will look for shipment records to purge.
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 are 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A-219 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_AUDIT
 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_HOLD_TYPE

YFS_SHIPMENT_HOLD_TYPE_LOG
YFS_SHIPMENT_TAG_SERIALS
YFS_SHIPMENT_LINE
YFS_SHIPMENT_LINE_REQ_TAG
YFS_ACTIVITY_DEMAND
YFS_CONTAINER_STS_AUDIT
YFS_CONTAINER_ACTIVITY

A.4.3.32 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 application server restart.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, the shipment statistics are picked up for purge:

- The last modification performed on the shipment statistics should fall before the lead time (in days) setup.

Attributes

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

Table A–220 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–221 *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 are 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.
ColonyID	Required in a multi schema deployment where the YFS_SHIPMENT_STATISTICS table may exist in multiple schemas. Runs the agent for the colony.

Statistics Parameters

The following are the statistics parameters for this transaction:

Table A–222 *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.4.3.33 Statistics Purge

This purge removes statistics data from the system. It purges all records older than the specified retention days.

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, the statistics detail is picked up for purge:

- The last modification performed on the statistics detail should fall before the lead time (in days) setup.

Note: This purge only reads the rules defined by the hub. Enterprise overridden rules are not considered. This purge should be single threaded when you run it in batch delete mode (BatchDelete=Y).

Note: Sterling Commerce 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–223 Statistics Purge Attributes

Attribute	Value
Base Transaction ID	STATTLPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–224 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 are moved to history tables without actually moving them.
BatchDelete	Required. The mode in which all records get deleted from the table. Valid values are: <ul style="list-style-type: none"> Y - Default value. Records are deleted in batches. N - Records are deleted one by one.

Table A–224 Statistics 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.
CollectPendingJobs	If this parameter is set to "N", the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
ColonyID	Required in a multi schema deployment where the YFS_STATISTICS_DETAIL table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–225 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.4.3.34 User Activity Purge

This purge deletes the user activity data from the system. It purges all records older than the specified retention days, and those records which

have a logged out status. This purge must be single threaded when you run it in batch delete mode (BatchDelete=Y).

The following limitation is assumed when purging records:

This purge do not purge any record if the Application server goes down abruptly because the audit records of users connected to the application server at the time when the server went down cannot be updated. As a result, the last activity time or the logout time is not populated. The purge does not know whether the user has logged out or still logged in. Therefore, you need to manually delete these records.

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

Table A–226 User Activity Purge Attributes

Attribute	Value
Base Transaction ID	USERACTIVITYPRG
Base Document Type	None
Base Process Type	None
APIs Called	None
User Exits Called	None

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–227 User Activity Purge 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.

Table A–227 User Activity Purge 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 are moved to history tables without actually moving them.
CollectPendingJobs	If this parameter is set to "N", the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
Number of Records To Buffer	Required. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 100.
BatchDelete	Required. The method by which all records are deleted from the table. Valid values are: <ul style="list-style-type: none"> Y - Default value. Records are deleted in batches. N - Records are deleted one by one.
ColonyID	Required in a multi schema deployment where the YFS_USER_ACTIVITY table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–228 Statistics Purge Statistics

Statistic Name	Description
NumStatisticsPurged	Number of statistics purged

Pending Job Count

None.

Events Raised

None.

Tables Purged

YFS_USER_ACTIVITY

A.4.3.35 User Activity Audit Purge

This purge removes user activity audit data from the system. It purges all records older than the specified retention days. It purges only those records which have a logged out status (records with a Login_Type of 'T' or 'N'). This purge should be single threaded when you run it in batch delete mode (BatchDelete=Y).

The following limitation is assumed when purging records:

- This purge does not purge any records if the Application server goes down abruptly because the audit records of users connected to application servers at the time the server went down cannot be updated. As a result, the last activity time or the logout time does not get populated and the purge does not know whether the user was logged out or was still logged in. These records have to be deleted manually.

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

Table A-229 User Activity Audit Purge Attributes

Attribute	Value
Base Transaction ID	USERACTAUDPURGE
Base Document Type	None
Base Process Type	None
APIs Called	None
User Exits Called	None

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–230 User Activity Audit 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. 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 are moved to history tables without actually moving them.
CollectPendingJobs	If this parameter is set to "N", the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
Number of Records To Buffer	Required. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 100.
BatchDelete	Required. The method by which all records are deleted from the table. Valid values are: <ul style="list-style-type: none"> • Y - Default value. Records are deleted in batches. • N - Records are deleted one by one.
ColonyID	Required in a multi schema deployment where the YFS_USER_ACT_AUDIT table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–231 Statistics Purge Statistics

Statistic Name	Description
NumStatisticsPurged	Number of statistics purged

Pending Job Count

None.

Events Raised

None.

Tables Purged

YFS_USR_ACT_AUDIT

A.4.3.36 Work Order History Purge

This transaction deletes tasks previously archived by the Work Order Purge. See [Section A.4.3.37, "Work Order Purge"](#).

You can use purge codes pseudo-logic to analyze purges. If the following condition is met, a work order that is previously purged by the work order purge agent is picked up for history purge:

- The last modified date of the work order should exceed the retention day period.

Attributes

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

Table A–232 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

Table A–232 Work Order History Purge Attributes

Attribute	Value
APIs Called	None
User Exits Called	YFSBeforePurgeUE

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–233 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 are 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.

Table A–233 Work Order History Purge Criteria Parameters

Parameter	Description
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that 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 Application Platform > System Administration > Agent Criteria Groups.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–234 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_AUDIT

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.4.3.37 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.

You can use purge codes pseudo-logic to analyze purges. If the following conditions are met, a work order is picked up for purge:

- The last modified date of the work order should exceed the retention day period.
- The order associated with the work order should be purged.
- The work order should be in pickable status by the purge transaction.

Attributes

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

Table A–235 Work Order Purge Attributes

Attribute	Value
Base Transaction ID	WORK_ORDER_PURGE
Base Document Type	Work Order
Base Process Type	VAS
Abstract Transaction	No

Table A–235 Work Order Purge Attributes

Attribute	Value
APIs Called	None
User Exits Called	YFSBeforePurgeUE

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–236 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 are 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.
Node	Optional. Node for which the Work Order Purge needs to be run. If not passed, then all nodes are monitored.

Table A–236 Work Order Purge Criteria Parameters

Parameter	Description
AgentCriteriaGroup	Optional. Used to classify nodes. This value can be accepted by WMS time-triggered transactions that 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 Application Platform > System Administration > Agent Criteria Groups.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–237 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_AUDIT

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.4.3.38 YFS Audit Purge

This purge removes the YFS_AUDIT table data from the system, which reduces the load on frequently accessed tables. It purges records in the YFS_AUDIT and the YFS_AUDIT_HEADER tables that meet the following conditions:

- YFS_AUDIT records that have 'modifyts' greater than the retention days specified and the value of table name matches in the YFS_AUDIT table.
- The last modified time is before the lead time (in days) setup.

Note: The way you configure the YFS Audit Purge may have some effect on the functioning of the Configuration Data Versioning Tool. For more information about configuration of the Data Versioning Tool, see the *Selling and Fulfillment Foundation: Configuration Deployment Tool Guide*.

When the enterprise extends the entities and sets the extended entities attribute AuditTable="Y", the extended tables are audited and the audit records are inserted in the YFS_AUDIT table. In order to clean up the audit records, this purge transaction can be used.

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–238 YFS Audit Purge Attributes

Attribute	Value
Base Transaction ID	YFS_AUDIT_PURGE
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–239 YFS Audit Purge Criteria Parameters

Parameter	Description
Action	Required. Triggers the transaction. If left blank, this value 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), this value defaults to 5000.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> Y - Default value. Production mode. Deletes records from the regular tables. N - Test mode.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
Table Name	Required. The table name for which the audit records need to be purged.

Table A–239 YFS Audit Purge Criteria Parameters

Parameter	Description
TableType	<p>Required in a multischema deployment when YFS_AUDIT table may exist in multiple schemas.</p> <p>Valid Values: CONFIGURATION, TRANSACTION, MASTER.</p> <p>If set to CONFIGURATION, the agent runs for the YFS_AUDIT records associated with tables that have TableType as CONFIGURATION; for example, YFS_Organization, YFS_Ship_Node, and so forth.</p> <p>If set to TRANSACTION, the agent runs for the YFS_AUDIT records associated with tables that have TableType as TRANSACTION; for example, YFS_Order_Header, YFS_Shipment, and so forth.</p> <p>Note that the agent would run for all TableTypes that exist in the same schema as the one passed. For example, if set to TRANSACTION, the agent would also run for YFS_AUDIT records associated with tables that have TableType as MASTER, since they reside in the same schema.</p>
ColonyID	<p>Required in a multi schema deployment where the YFS_AUDIT and YFS_AUDIT_HEADER tables may exist in multiple schemas. Runs the agent for the colony.</p>

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–240 YFS Audit Purge Statistics

Statistic Name	Description
NumAuditRecordsPurged	Number of audit records purged.

Pending Job Count

For this transaction, the pending job count is the number of records that can be purged from the YFS_AUDIT table that match the criteria values.

Events Raised

None.

Tables Purged

YFS_AUDIT, YFS_AUDIT_HEADER

A.4.3.39 YFSInventoryOwnershipAudit Purge

This transaction purges all the records from YFS_INV_OWN_TRANSFER_RCD prior to the lead days specified in criteria parameters.

Attributes

Following are the attributes for this time-triggered transaction:

Table A–241 YFSInventoryOwnership Purge Attributes

Attribute	Value
Base Transaction ID	PURGE_INV_TRANSFR_RECORD
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	None

Criteria Parameters

Following are the criteria parameters for this transaction:

Table A–242 YFSInventoryOwnership Purge Criteria Parameters

Parameter	Description
Action	Required. Triggers the transaction. If left blank, this value defaults to Get, which is 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), this value defaults to 5000.

Table A–242 YFSInventoryOwnership Purge Criteria Parameters

Parameter	Description
EnterpriseCode	Optional. The inventory organization for which the YFSInventoryOwnership Audit Purge needs to run. If not passed, all the enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> • Y - Default value. Production mode. Deletes records from the regular tables. • N - Test mode.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds to the PurgeCode used in the Business Rules Purge Criteria.
Lead Days	Number of days before the present date, the agent will purge the records.
ColonyID	Required in a multi schema deployment where the YFS_INV_OWN_TRANSFER_RCD table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

None.

Tables Purged

YFS_INV_OWN_TRANSFER_RCD

A.4.3.40 Password Reset Request Purge

This purge deletes password reset request data from the system.

You can use purge codes pseudo-logic to analyze purges.

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–243 Password Reset Request Purge Attributes

Attribute	Value
Base Transaction ID	None
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–244 Password Reset Request 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 are moved to history tables without actually moving them.

Table A–244 Password Reset Request 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.
ColonyID	Required in a multi schema deployment where the PLT_PWD_REQ table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–245 Password Reset Request Purge Statistics

Statistic Name	Description
NumPasswordRequestPurged	Number of password requests purged.

Pending Job Count

For this transaction, the pending job count is the number of records that can be purged from the PLT_PWD_REQ table.

Events Raised

None.

Tables Purged

PLT_PWD_REQ

A.4.3.41 User Login Failure Purge

This purge deletes data on number of failed login attempts of users from the system.

You can use purge codes pseudo-logic to analyze purges.

Any enterprise using the Console must schedule purge transactions.

Attributes

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

Table A–246 *User Login Failure Purge Attributes*

Attribute	Value
Base Transaction ID	None
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–247 *User Login Failure 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 are moved to history tables without actually moving them.

Table A–247 *User Login Failure 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.
ColonyID	Required in a multi schema deployment where the PLT_USER_LOGIN_FAILED table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–248 *User Login Failure Purge Statistics*

Statistic Name	Description
NumUserLoginFailPurged	Number of failed login attempts purged.

Pending Job Count

For this transaction, the pending job count is the number of records that can be purged from the PLT_USER_LOGIN_FAILED table.

Events Raised

None.

Tables Purged

PLT_USER_LOGIN_FAILED

A.5 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 synchroners 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 8.5 for time-triggered transactions, monitors, and integration and application servers may change with the next release.

A.5.1 Load Execution Task Queue Syncher

This transaction synchronizes the task queue for the load execution process type.

You can use the following pseudo-logic to analyze this time-triggered transaction. If the following conditions are met, a task queue for the load execution process type is synchronized:

- `LOAD_CLOSED_FLAG` of Load should not be 'Y'.
- Load should be in a status that is pickable by a transaction in the pipeline.
- There should not be any Task Q record for the load, transaction combination in the Task Q table. In this case, the system inserts one Task Q record for this load, transaction combination with the current database time as the available date.

Attributes

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

Table A–249 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–250 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–251 Load Execution Task Queue Syncher Statistics

Statistic Name	Description
NumTasksCreated	Number of tasks created.

Pending Job Count

None.

Events Raised

None.

A.5.2 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–252 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–253 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–254 Order Delivery Task Queue Syncher Statistics

Statistic Name	Description
NumTasksCreated	Number of tasks created.

Pending Job Count

None.

Events Raised

None.

A.5.3 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–255 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
Abstract Transaction	No
APIs Called	None

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–256 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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–257 Order Fulfillment Task Queue Syncher Statistics

Statistic Name	Description
NumTasksCreated	Number of tasks created.

Pending Job Count

None.

Events Raised

None.

A.5.4 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–258 *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–259 *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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–260 *Order Negotiation Task Queue Syncher Statistics*

Statistic Name	Description
NumTasksCreated	Number of tasks created.

Pending Job Count

None.

Events Raised

None.

A.6 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 8.5 for time-triggered transactions, monitors, and integration and application servers may change with the next release of Selling and Fulfillment Foundation.

Note: All Monitors have a CollectPendingJobs criteria parameter. If this parameter is set to N, the agent does not collect information on the pending jobs for that monitor. This pending job information is used for monitoring the monitor in the System Management Console. 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.6.1 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–261 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–262 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.

Table A–262 Availability Monitor Criteria Parameters

Parameter	Description
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
Status	The negotiation status you are monitoring.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

None.

Events Raised

No events are raised. Individual actions associated with the monitoring rule are run.

Data published to the actions is `AVAILABILITY_MONITOR_dbd.txt`.

A.6.2 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 Applications Manager. 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–263 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–264 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.
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.

Table A–264 Exception Monitor Criteria Parameters

Parameter	Description
QueueGroup	Optional. Defines the set of Queues for which the exceptions will be monitored. If both QueueId and QueueGroup are supplied, QueueId is ignored.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–265 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 run.

A.6.3 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 *Sterling Global Inventory Visibility: Configuration Guide*.

Attributes

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

Table A–266 *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–267 *Inventory 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.

Table A–267 Inventory Monitor Criteria Parameters

Parameter	Description
InventoryOrganizationCode	Optional. Valid inventory owner organization. Organization to process in this run. If not passed, all inventory organizations are processed.
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
AllowedOverriddenCriteria	If this parameter is set to Y, the overriding value for the agent criteria parameters can be provided in the command line in the following format when triggering the agent: <AgentCriteriaAttribute> <OverriddenValue> For more information about passing these attributes, see the <i>Selling and Fulfillment Foundation: Installation Guide</i>
ShipNodes	Optional. Comma-separated list of valid ship nodes that should be processed in this run. If not passed, all the ship nodes are processed.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

None.

Events Raised

No events are raised. Individual actions associated with the monitoring rule are run.

Data published to the actions is `<INSTALL_DIR>/xapidocs/api_javadocs/dbd/INVENTORY_MONITOR_dbd.txt`.

A.6.4 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-268 *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–269 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.
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
Status	The negotiation status you are monitoring.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–270 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.6.5 Enhanced Order Monitor

The enhanced order monitor enables 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 has been in hold type x for y hours.
- The order has been in hold type x for y hours before 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.
- Actual Next Iteration Date - Read from the NEXT_ITER_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.
- 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 *Sterling Supply Collaboration: Configuration Guide*, the *Sterling Distributed Order Management: Configuration Guide*, and the *Sterling 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 information about the Close Order time-triggered transaction, see [Section A.3.8, "Close Order"](#).

Note: The same relog interval is used for all document types.

Attributes

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

Table A–271 *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–272 *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.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this monitor:

Table A–273 Enhanced Order Monitor Statistics

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumAlertsRaised	Number of alerts raised.

Pending Job Count

For this transaction the pending job count is the number of open orders with the value of NEXT_ALERT_TS less than or equal to (<=) the current date.

Events Raised

Table A–274 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: <INSTALL_DIR>/xapidocs/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.

Monitor Rule's Condition Template

If a monitor rule contains a condition, the <INSTALL_DIR>/repository/xapi/template/source/smcfs/monitor/ORDER_MONITOR_EX_CONDITION.xml template file is used to obtain both the order details and the evaluating monitor rule details. See the provided <INSTALL_

DIR>/repository/xapi/template/source/smcfs/monitor/ORDER_MONITOR_EX_CONDITION.xml.sample file for more details.

If the <INSTALL_

DIR>/repository/xapi/template/source/smcfs/monitor/ORDER_MONITOR_EX_CONDITION.xml template file does not exist, the MonitorConsolidation->Order element of the default monitor template, the <INSTALL_

DIR>/repository/xapi/template/source/smcfs/monitor/ORDER_MONITOR_EX.xml file, is used.

Note: Note: If the default monitor template is used, the MonitorConsolidation-> Order->OrderStatuses-> OrderStatus->MonitorRule element is ignored and is not passed into the condition.

A.6.6 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 *Sterling Supply Collaboration: Configuration Guide*, the

Sterling Distributed Order Management: Configuration Guide, and the *Sterling 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 information about the Close Order time-triggered transaction, see [Section A.3.8, "Close Order"](#).

Note: The same relog interval is used for all document types.

Attributes

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

Table A–275 *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–276 *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.

Table A–276 Enhanced Order Monitor Criteria Parameters

Parameter	Description
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.
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this monitor:

Table A–277 Enhanced Order Monitor Statistics

Statistic Name	Description
NumOrdersProcessed	Number of orders processed.
NumAlertsRaised	Number of alerts raised.

Pending Job Count

For this transaction the pending job count is the number of open orders with the value of NEXT_ALERT_TS less than or equal to (<=) the current date.

Events Raised

No events are raised. Individual actions associated with the monitoring rule are run.

The data published is RETURN_MONITOR_EX.xml.

Monitor Rule's Condition Template

If a monitor rule contains a condition, the `<INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/ORDER_
MONITOR_EX_CONDITION.xml` template file is used to obtain both the order details and the evaluating monitor rule details. See the provided `<INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/ORDER_
MONITOR_EX_CONDITION.xml.sample` file for more details.

If the `<INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/ORDER_
MONITOR_EX_CONDITION.xml` template file does not exist, the `MonitorConsolidation->Order` element of the default monitor template, the `<INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/ORDER_
MONITOR_EX.xml` file, is used.

Note: Note: If the default monitor template is used, the `MonitorConsolidation-> Order->OrderStatuses-> OrderStatus->MonitorRule` element is ignored and is not passed into the condition.

A.6.7 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 between the thresholds defined in the Applications Manager in the Global Inventory Visibility 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 is used for considering inventory availability at retrieval time. For more information

about future inventory availability, see the *Sterling Global Inventory Visibility: Configuration Guide*.

Inventory available at the current date is considered as on-hand. The processing time in the ATP rules must be set to at least 1 day, or else past due supply is included as part of on-hand inventory. For more information about configuring ATP Rules, see the *Sterling Global Inventory Visibility: Configuration Guide*.

Demand of type OPEN_ORDER is 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 about configuring distribution groups and node-level inventory monitoring, see the *Sterling Global Inventory Visibility: Configuration Guide*.

Inventory items without an Availability Monitor rule, or with a rule that is disabled, is unable to be processed by this time-triggered transaction.

If configured, the Real-time Availability Monitor also considers the onhand and future inventory availability safety factor during monitoring. For more information about the inventory availability safety factors and the `findInventory()` API, see the *Sterling Global Inventory Visibility: Configuration Guide* and the *Selling and Fulfillment Foundation: Javadocs*.

When the onhand quantity is greater than the configured low threshold, the `REALTIME_ONHAND` alert type is raised, and the alert level is based on the onhand quantity.

When the onhand 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 `FirstFutureAvailableDate` 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 does 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 is 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–278 *Real-time Availability Monitor Attributes*

Attribute	Value
Base Transaction ID	REALTIME_ATP_MONITOR
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	FindInventory

Criteria Parameters

The following are the criteria parameters for this monitor:

Table A–279 Real-time Availability Monitor Criteria Parameters

Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
InventoryOrganizationCode	Inventory organization code to use when MonitorOption is passed as 3. The inventory organization has to be an enterprise. If this is not passed, the monitor runs for all inventory organizations.
MonitorOption	1 - Activity Based (Monitor based on distinct inventory items in YFS_INVENTORY_ACTIVITY table). 2 – Quick Sync (Re-raise event to publish information from the YFS_INVENTORY_ALERT table). 3 – Full Sync (Monitor based on all inventory items maintained by the inventory organization provided. If no InventoryOrganizationCode is provided, all inventory item is monitored). If not provided, default value is 1.
ItemStatuses	List of valid statuses of items to be processed. Statuses must be separated by a , for example 3000,2000. This is only used when MonitorOption is passed as 2 or 3. If provided, only items with the matching statuses is monitored.

Table A–279 Real-time Availability Monitor Criteria Parameters

Parameter	Description
FromAlertTimestamp	<p>This is only used when MonitorOption is passed as 2. If provided, the agent raises 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 is 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="762 786 1143 838"><AgentCriteriaAttribute> <OverriddenValue></pre> <p>For more information about passing these attributes, see the <i>Selling and Fulfillment Foundation: Installation Guide</i>.</p>
FromLastNumberOfHours	<p>This is only 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>
CollectPendingJobs	<p>If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.</p>

Table A–279 Real-time Availability Monitor Criteria Parameters

Parameter	Description
RaiseEventsOnAllAvailability Changes	When set to Y, REALTIME_AVAILABILITY_CHANGE event is raised on all availability changes regardless of whether availability exceeds or falls below specified thresholds. This is only used when MonitorOption is passed as 1. Valid values: Y or N. Default value: N.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

None.

Pending Job Count

None.

Events Raised

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

Table A–280 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: <INSTALL_DIR>/xapidocs/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.6.8 Shipment Monitor

This time-triggered transaction reports the states of a shipment, based on rules in the YFS_MONITOR_RULE table. This transaction enables 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
- If the Shipment has been in a hold type for a specified amount of time.
- If the Shipment has been in a hold type for n hours before a 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 *Sterling Supply Collaboration: Configuration Guide*, the *Sterling Distributed Order Management: Configuration Guide*, and the *Sterling Reverse Logistics: Configuration Guide*.

Attributes

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

Table A–281 *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–282 *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.

Table A–282 Shipment Monitor Criteria Parameters

Parameter	Description
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–283 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 (<=) the current date.

Events Raised

This invokes the actions configured against shipment statuses.

Key Data - Not Applicable.

Data Published - SHIPMENT_MONITOR.xml

Monitor Rule's Condition Template

If a monitor rule contains a condition, the <INSTALL_DIR>/repository/xapi/template/source/smcfs/monitor/SHIPMENT_MONITOR_CONDITION.xml template file is used to obtain the shipment details and the evaluating monitor rule details. See the provided <INSTALL_DIR>/repository/xapi/template/source/smcfs/monitor/SHIPMENT_MONITOR_CONDITION.xml.sample file for more details.

If the <INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/SHIPMENT_
MONITOR_CONDITION.xml template file does not exist, the
MonitorConsolidation->Shipment element of the default monitor
template, the <INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/SHIPMENT_
MONITOR.xml file, is used.

Note: Note: If the default monitor template is used, the
MonitorConsolidation->Shipment->MonitorRule element is
ignored and is not passed into the condition.

A.6.9 Work Order Monitor

This time-triggered transaction alerts the enterprise when a work order remains in a particular state or hold type for a specific amount of time.

Use this monitor to track how long work orders stay in a particular state or hold type.

Attributes

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

Table A–284 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–285 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.
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
ColonyID	Required in a multi schema deployment where a table may exist in multiple schemas. Runs the agent for the colony.

Statistics Tracked

The following statistics are tracked for this transaction:

Table A–286 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 run. Data published to the actions is workOrder_dbd.txt.

Monitor Rule's Condition Template

If a monitor rule contains a condition, the `<INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/monitor/WOR
K_ORDER_MONITOR_CONDITION.xml` template file is used to obtain the
work order details and the evaluating monitor rule details. See the
provided `<INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/WORK_ORDER_
MONITOR_CONDITION.xml.sample` file for more details.

If the `<INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/WORK_ORDER_
MONITOR_CONDITION.xml` template file does not exist, the
MonitorConsolidation->WorkOrder element of the default monitor
template, the `<INSTALL_
DIR>/repository/xapi/template/source/smcfs/monitor/WORK_ORDER_
MONITOR.xml` file, is used.

Note: If the default monitor template is used, the MonitorConsolidation->WorkOrder->->MonitorRule element is ignored and is not passed into the condition.

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 Selling and Fulfillment Foundation. It publishes detailed information about the changed supply.

Key Data: `INSTALL_DIR/xapidocs/api_javadocs/dbd/INVENTORY_CHANGE_SUPPLY_CHANGE_dbd.txt`

Data Published: `INSTALL_DIR/xapidocs/api_javadocs/XSD/HTML/INV_INVENTORY_CHANGE_SUPPLY_CHANGE.html`

- **DEMAND_CHANGE**

This event is raised for every inventory demand change within Selling and Fulfillment Foundation. It publishes detailed information about the changed demand. Order information is published when the inventory organization is set to create demand details.

Key Data: `INSTALL_DIR/xapidocs/api_javadocs/dbd/INVENTORY_CHANGE_DEMAND_CHANGE_dbd.txt`

Data Published: `INSTALL_DIR/xapidocs/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 Selling and Fulfillment Foundation.

Key Data: `INSTALL_DIR/xapidocs/api_javadocs/dbd/INVENTORY_CHANGE_EXTERNAL_SUPPLY_CHANGE_dbd.txt`

Data Published: INSTALL_DIR/xapidocs/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 Selling and Fulfillment Foundation.

Key Data: INSTALL_DIR/xapidocs/api_javadocs/dbd/INVENTORY_CHANGE_EXTERNAL_DEMAND_CHANGE_dbd.txt

Data Published: INSTALL_DIR/xapidocs\api_javadocs\XSD\HTML\INV_INVENTORY_CHANGE_EXTERNAL_DEMAND_CHANGE.html

- INVENTORY_CHANGE

This event is raised only when we adjust Inventory with ONHAND_SUPPLY supply type. For example, when Inventory changes from HELD to ONHAND supply. It publishes detailed information of the changed supply.

Key Data: INSTALL_DIR/xapidocs/api_javadocs/dbd/INVENTORY_CHANGE_INVENTORY_CHANGE_dbd.txt

Data Published: INSTALL_DIR/xapidocs/api_javadocs/dbd/INVENTORY_CHANGE_INVENTORY_CHANGE_dbd.txt

- ON_INV_OWNERSHIP_TRNSFR

This event is raised each time a transfer of inventory ownership happens. It publishes the information about the transfer of inventory ownership.

Key Data: INSTALL_DIR/xapidocs/api_javadocs/dbd/INVENTORY_CHANGE_INVENTORY_TRANSFER_dbd.txt

Data Published - INSTALL_DIR/xapidocs/api_javadocs/XSD/HTML/INV_INVENTORY_CHANGE.ON_INV_OWNERSHIP_TRANSFER.html

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

Table B–2 Capacity Change Attributes

Attribute	Value
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: `INSTALL_DIR/xapidocs/api_javadocs/dbd/ CAPACITY_CHANGE_ON_CAPACITY_ADJUSTMENT_dbd.txt`

Data Published: `INSTALL_DIR/xapidocs/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: `INSTALL_DIR/xapidocs/api_javadocs/dbd/ CAPACITY_CHANGE_ON_OVERRIDE_CAPACITY_dbd.txt`

Data Published: `INSTALL_DIR/xapidocs/api_javadocs/XSD/HTML/INV_CAPACITY_CHANGE.ON_OVERRIDE_CAPACITY.xml`

- **EXTERNAL_CAPACITY_CHANGE**

This event is raised for every external capacity change within Selling and Fulfillment Foundation. It publishes detailed information about the changed capacity of external resource pools.

Key Data: `INSTALL_DIR/xapidocs/api_javadocs/dbd/ CAPACITY_CHANGE_EXTERNAL_CAPACITY_CHANGE_dbd.txt`

Data Published: `INSTALL_DIR/xapidocs/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, MQSeries, and TIBCO JMS Queue](#)
- [Microsoft Message Queue \(MSMQ\)](#)
- [WebSphere Default Messaging JMS Queue](#)

- [JBoss Default Messaging JMS Queue](#)

Exception Handling

Asynchronous receiver links can be configured to handle exceptions, as described in [Section C.1.15, "Receiver Link Exception Handling"](#) on page C-847. The asynchronous receiver links that can be configured for exception handling are the following:

- [Database Receiver](#)
- [File I/O Receiver](#)
- [FTP Receiver](#)
- [WebLogic, MQSeries, and TIBCO 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: It is recommended that you do not use the database node when scaling. Instead, use a JMS Queue node.

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.
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 Selling and Fulfillment Foundation 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	

Table C-1 Database Sender Configuration Properties

Property	Description
	<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>In addition to the Header Name/Value, EOF messages get appended with MessageType=EOF in the USER_REFERENCE column. If you are using USER_REFERENCE as the selector, you need to modify the Selector criteria to include MessageType=EOF as well.</p> <p>Note: The header name must be unique. Two headers cannot have the same header name.</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>Selling and Fulfillment Foundation: Application 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 USER_REFERENCE='<name>=value'</p> <p>In addition to the Header Name/Value, EOF messages get appended with MessageType=EOF in the USER_REFERENCE column. If you are using USER_REFERENCE as the selector, you need to modify the Selector criteria to include MessageType=EOF as well.</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 FLOW_NAME, SUB_FLOW_NAME, USER_REFERENCE, from the YFS_EXPORT/IMPORT table.</p>

Table C–2 Database Receiver Configuration Properties

Property	Description
Table Name	<p>Select the table you want the message to be read from. Valid values are YFS_IMPORT and YFS_EXPORT.</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 End Of File (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>customer_overrides.properties</code> file) before executing this service. For more information see, Section C.1.5.3, "Enabling EOF Messages in the Application Platform Framework". For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
Root Node Name of EOF Message	<p>This need to be specified only if the message contains an EOF message ID.</p> <p>Enter your custom root node name for the EOF message.</p> <p>By default the EOF message has a root node name as "EOF". For more information see, Section C.1.5.3, "Enabling EOF Messages in the Application Platform Framework"</p>
Server Tab	

Table C–2 Database Receiver Configuration Properties

Property	Description
Server Name	Required. Name of the integration server instance which actually executes the service. For more information about creating a new server, see "Adding a New Server" .
Exception Tabs	See Table C–37 for the receiver link exception handling properties.

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 about 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 <code>UPLD</code> or <code>DNLD</code> 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 <code>UPLD</code> .
Node	Enter the ship node associated with the service.

Table C-4 DCS 6.2 Database Sender Configuration Properties

Property	Description
Table Name	<p>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.</p> <p>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.</p>
Remote Host ID	Enter the remote host ID in the database.
Rollback on Exception	<p>Check this box if you want to rollback the database write if the parent transaction fails.</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 Selling and Fulfillment Foundation 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.</p>

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	
Sub Service Name	Enter a unique identifier for each asynchronous receiver.

Table C–5 DCS 6.2 Database Receiver Configuration Properties

Property	Description
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 about creating a new server, see "Adding a New Server" .
Exception Tabs	See Table C–37 for the receiver link exception handling properties.

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 <INSTALL_DIR>/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 other, non-Selling and Fulfillment FoundationEJBs.

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.
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.

Table C–9 EJB Sender Configuration Properties

Property	Description
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	<p>Enter the directory into which the incoming files are placed before processing begins.</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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
Completion Directory	<p>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.</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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
File Prefix	<p>Enter the name to prefix to each file name. Used to differentiate output files from different services.</p>

Table C–11 File I/O Sender Connection Properties

Property	Description
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.
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 Application Platform default for the JVM unless otherwise specified.
Max Time to Rollover (mins)	Enter the maximum time (in minutes) the output file remains open.
Disable Root Element	<p>If you check this box, based on the value of the Max Transactions Per File property, the XMLEMENT root element is deleted from the XML file.</p> <ul style="list-style-type: none"> • If the value of the Max Transactions Per File property is equal to 1, the XMLEMENT root element is deleted from the XML file. • If the value of the Max Transactions Per File property is greater than 1, the XMLEMENT root element is not deleted from the XML file.

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 <code>posCreateOrder.*</code> 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 Application 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.
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.

Table C–12 File I/O Receiver Configuration Properties

Property	Description
Create EOF Message	Select this field if you want to create 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 about creating EOF messages, see Section C.1.5.3, "Enabling EOF Messages in the Application Platform Framework" .
Do Not Fragment	<p>If you check this box, the output XML file is not fragmented into 'n' number of separate files. For example, for a load, if you create two different load lines, the output XML file contains the "Load" root element and two "LoadList" child elements.</p> <p>If you do not check this box, the root element is deleted from the output XML file and a separate XML file is created for each of the child elements.</p>
File Tab	
Incoming Directory	<p>Enter the directory in which to look for input files.</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 the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file. For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>

Table C–12 File I/O Receiver Configuration Properties

Property	Description
Completion Directory	<p>Enter the directory into which the processed files are archived after they have been processed.</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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about modifying properties and the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Installation Guide</i>.</p>
Working Directory	<p>Enter the directory into which the files from the incoming directory are moved before processing begins.</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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>

Table C–12 File I/O Receiver Configuration Properties

Property	Description
Error Directory	<p>Enter the directory into which any errors in the file being processed are created. A new error file is created for each file processed.</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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
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 .err.
Completion File Suffix	Required if the Completion Directory is specified. Defaults to .done.
Server Tab	
Server Name	<p>Enter the name of the integration server instance which actually executes the service.</p> <p>For more information about creating a new server, see "Adding a New Server".</p>
Exception Tabs	See Table C–37 for the receiver link exception handling properties.

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 Application 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 the Selling and Fulfillment Foundation File IO adapter component, 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 Selling and Fulfillment Foundation:

- 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 Selling and Fulfillment Foundation 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 the `<INSTALL_DIR>/properties/customer_overrides.properties` file, before calling the service.

For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

In general, the XML file passed to the service must have a root node name as 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.

For example, when you customize the root node name as `Root Node Name Of EOF Message: NewRootNode` and `Service to execute on EOF Message: NewService`, add the XML attributes that must be included in the input to the `NewService` with the with the `MessageType` value as "EOF" as follows:

```
<EOF YantraMessageGroupID='123' />
```

The input to the `NewService` would be constructed as `<NewRootNode YantraMessageGroupID='123' />`

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.

Table C–14 FTP Sender Configuration Properties

Attribute	Description
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	<p>Enter the regular expression pattern of the files to retrieve. If not specified all files in the working directory are included.</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.
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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i> .
Source File Parameters Tab	

Table C–14 FTP Sender Configuration Properties

Attribute	Description
Working Directory	<p>Enter the directory on the FTP server from which files are 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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
Completion Directory	<p>Enter the remote completion directory files are to be moved to after they are successfully transferred from the remote staging 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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
File Separator	<p>Enter the file separator used on the FTP server. The file separator is the folder delimiter specified by the FTP server's settings. This is most commonly represented as a forward slash ('/'). This field is mandatory.</p>
Completion Suffix	<p>Enter the suffix to append the file when the file from the remote working directory is moved into the remote completion directory.</p>
Destination File Properties Tab	

Table C–14 FTP Sender Configuration Properties

Attribute	Description
Working Directory	<p>Enter the directory in which the files received are transferred to.</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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
Completion Directory	<p>Enter the local completion directory files are to be moved to after they are successfully transferred from the remote working 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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
File Separator	<p>Enter the file separator used on the FTP server. This field is mandatory.</p>
Completion Suffix	<p>Enter the suffix to append the file when the file from the local working directory is moved into the local completion directory.</p>
Server Tab	

Table C–14 FTP Sender Configuration Properties

Attribute	Description
Server Name	Enter the name of the server which actually executes the service. For more information about creating a new server, see " Adding a New Server ".
Exception Tabs	See Table C–37 for the receiver link exception handling properties.

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.
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.

Table C–15 FTP Receiver Configuration Properties

Attribute	Description
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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>

Table C–15 FTP Receiver Configuration Properties

Attribute	Description
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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
File Separator	<p>Enter the file separator used on the FTP server. The file separator is the folder delimiter specified by the FTP server's settings. This is most commonly represented as a forward slash ('/'). This field is mandatory.</p>
Completion Suffix	<p>Enter the suffix to append the file when the file from the remote working directory is moved into the remote completion directory.</p>
Destination File Properties Tab	
Working Directory	<p>Enter the directory in which the files received are transferred to.</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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>

Table C–15 FTP Receiver Configuration Properties

Attribute	Description
Completion Directory	<p>Enter the completion directory. After a file is transferred successfully from the remote working directory, it is subsequently moved to the local 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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
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	<p>Enter the name of the server which actually executes the service.</p> <p>For more information about creating a new server, see "Adding a New Server".</p>
Exception Tabs	See Table C–37 for the receiver link exception handling properties.

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.
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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties 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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties 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.rsa.jca.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

Table C–18 HTTP Connection Properties

Connection	Node Connection Rules
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 the `smcfs.ear`, see the *Selling and Fulfillment Foundation: Installation Guide*.

Note: Webservice SDF component is deprecated and will be removed in the future version. It is recommended that you do not use the WebServices SDF component to invoke WebServices. Instead, create a custom API to invoke WebServices.

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:

Configurable depending on your app server

Web log we always document literal

websphere

For WebSphere,

Pass `-D websphere-java2wsdl-style=<rpc|document>` in the ear command line

For JBoss,

Pass `-D jboss-java2wsdl-style=<rpc|document>` in the ear command line.

For WebLogic – we always use document literal.

JBoss based on variable

Build Ear.sh documentation where it talks about style of coding

WebLogic and Websphere are supported and paramizable and the other supported for document literal

Table C–19 WebServices Configuration Properties

Property	Description
General Tab	
URL	Enter the URL to which the message is to be posted. For example, <code>http://localhost:7001/smcfseljbservice</code>
Target Object URN	Enter the web service's resource name. For example, <code>yantrawebsevice</code>
Are Selling and Fulfillment Foundation Webservice	Check this box to indicate that this service is calling a Selling and Fulfillment Foundation Webservice. If you check this box, 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. For example, <code>http://schemas.xmlsoap.org/soap/encoding/</code>
Method Name	Enter the name of the method you want to invoke.

Table C–19 WebServices Configuration Properties

Property	Description
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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i> .
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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties 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, http://tempuri.org/PricingEngineGold/Service1/PricingEngineFunc can be a valid input URI.
Arguments Tab	

Table C–19 WebServices Configuration Properties

Property	Description
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
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>.

Table C–21 Synchronous WebLogic JMSQueue and MQSeries Properties

Property	Description
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 configure Queue and/or JNDI based JMS security.</p> <p>Note: If JMS security is enabled, the JMS session pooling should be disabled. You can disable JMS session pooling by setting the <code>yfs.jms.session.disable.pooling</code> property to 'Y' in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file. For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>

Table C–21 Synchronous WebLogic JMSQueue and MQSeries Properties

Property	Description
Response Tab	
Queue Name	Enter the name of the queue in which the response message is received.
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='APINAME' and value='createOrder'.
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, APINAME.</p> <p>Note: The header name must be unique. Two headers cannot have the same header name.</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>

Table C–21 Synchronous WebLogic JMSQueue and MQSeries Properties

Property	Description
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>
Reconnect Tab	
Retry Interval (milliseconds)	<p>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.</p>
Number of Retries	<p>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.</p>
JMS Security Properties Tab	
<p>This is enabled upon selecting Enable JMS Security in the runtime properties tab.</p> <p>Note: You can override the JMS security properties specified here by defining the agent and flow authorization parameters in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file. For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>	

Table C–21 Synchronous WebLogic JMSQueue and MQSeries Properties

Property	Description
Queue based security	Check this box, if you want to provide Queue based security for JMS service.
JNDI based security	Check this box, if you want to provide JNDI based security for JMS service.
JMS Security Parameters For more information about application server-specific JMS security parameters, 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 Application 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.

The `yfs.jms.session.reaptime` property in the `yfs.properties` file is used to set the JMS session reaptime. To modify this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

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, MQSeries, and TIBCO JMS Queue

This component provides a common interface to invoke messaging services on WebLogic, MQSeries, and TIBCO 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, MQSeries, and TIBCO 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, MQSeries, and TIBCO 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, MQSeries, and TIBCO 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.
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> • For TIBCO JMS, set to <code>tcp://<DNS Server Name or IP Address>:<port></code>

Table C–23 WebLogic, MQSeries, and TIBCO JMS Sender Configuration Properties

Property	Description
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> <p>Select TIBCO if you are using TIBCO JMS. This sets the class name to <code>com.tibco.tibjms.naming.TibjmsInitialContextFactory</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, MQSeries, or TIBCO.</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, MQSeries, and TIBCO 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 Selling and Fulfillment Foundation 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> <p>Note: If you check this box, the system synchronizes the JMS commit along with the DB commit. However, it is not a two-phase commit. Therefore, it may cause some errors, thus resulting in duplicate messages.</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 configure Queue and/or JNDI based JMS security.</p> <p>Note: If JMS security is enabled, the JMS session pooling should be disabled. You can disable JMS session pooling by setting the <code>yfs.jms.session.disable.pooling</code> property to 'Y' in the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file. For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>
Header Tab	

Table C–23 *WebLogic, MQSeries, and TIBCO JMS Sender Configuration Properties*

Property	Description
Header Name	<p>The name of the message header. For example, APINAME.</p> <p>Note: The header name must be unique. Two headers cannot have the same header name.</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>
<p>Reconnect Tab</p> <p>You need to use the settings in this tab are for the backup JMS server.</p>	
Retry Interval (milliseconds)	<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>

Table C–23 WebLogic, MQSeries, and TIBCO JMS Sender Configuration Properties

Property	Description
Number of Retries	<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>
Use backup JMS Queue	<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>
Provider URL	<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>:<bootstrapport></code> • For TIBCO JMS, set to <code>tcp://<DNS Server Name or IP Address>:<port></code>

Table C–23 WebLogic, MQSeries, and TIBCO 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>. • Select TIBCO if you are using TIBCO JMS. This sets the class name to <code>com.tibco.tibjms.naming.TibjmsInitialContextFactory</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, MQSeries, or TIBCO.
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.

Table C–23 WebLogic, MQSeries, and TIBCO JMS Sender Configuration Properties

Property	Description
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 reverts back to using the primary JMS server. If contact with the primary JMS server cannot be established, it continues to use the backup JMS server. The default value is 600 seconds.
<p>JMS Security Properties Tab</p> <p>This is enabled upon selecting Enable JMS Security in the runtime tab.</p> <p>Note: You can override the JMS security properties specified here by defining the agent and flow authorization parameters in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>	
Queue based security	Check this box, if you want to provide Queue based security for JMS service.
JNDI based security	Check this box, if you want to provide JNDI based security for JMS service.
<p>JMS Security Parameters</p> <p>For more information about application server-specific JMS security parameters, 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 Application 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.

The `yfs.jms.session.reaptime` property in the `yfs.properties` file is used to set the JMS session reaptime. To modify this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

C.1.10.2 WebLogic, MQSeries, and TIBCO 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, MQSeries, and TIBCO 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> • For TIBCO JMS, set to <code>tcp://<DNS Server Name or IP Address>:<port></code>

Table C–24 WebLogic, MQSeries, and TIBCO JMS Receiver Configuration Properties

Property	Description
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> <p>Select TIBCO if you are using TIBCO JMS. This sets the class name to <code>com.tibco.tibjms.naming.TibjmsInitialContextFactory</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, MQSeries, or TIBCO.</p>

Table C–24 WebLogic, MQSeries, and TIBCO JMS Receiver Configuration Properties

Property	Description
Receiving Mode	<p>Indicates whether the messages are received in a transactional mode or non-transactional mode.</p> <ul style="list-style-type: none"> • If you set this to non-transactional mode, the messages are removed from the queue as soon as it is read. • If you set this to transactional mode, the messages are removed from the queue when they are functionally processed and if an exception is thrown when processing it. <p>If the IsReprocessable flag is checked and an exception occurs when processing the message, then before deleting these messages, ensure that they are added to the YFS_REPROCESS_ERROR or YFS_INBOX table. However, if the "service suspend" exception occurs when adding these deleted messages to a table, the messages are maintained in the queue.</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>Selling and Fulfillment Foundation: System Manangement and Administration Guide</i>.</p> <p>You can also start multiple instances of the Service Definition Framework for a specific integration adapter for a specific server. For more information about the integration adapter, see the <i>Selling and Fulfillment Foundation: Performance Management Guide</i>.</p>

Table C–24 WebLogic, MQSeries, and TIBCO JMS Receiver Configuration Properties

Property	Description
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'.
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 <INSTALL_DIR>/properties/customer_overrides.properties file) before executing this service. For more information see, Section C.1.5.3, "Enabling EOF Messages in the Application Platform Framework" . For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i> .
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 Application 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 configure Queue and/or JNDI based JMS security.
Server Tab	

Table C–24 WebLogic, MQSeries, and TIBCO JMS Receiver Configuration Properties

Property	Description
Server Name	Select the name of the server which actually executes the service. For more information about creating a new server, see "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 Table C–37 for the receiver link exception handling properties.
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. Note: You can override the JMS security properties specified here by defining the agent and flow authorization parameters in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i> .	
Queue based security	Check this box, if you want to provide Queue based security for JMS service.
JNDI based security	Check this box, if you want to provide JNDI based security for JMS service.
JMS Security Parameters	
For more information about application server-specific JMS security parameters, 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.

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 WebSphere Default Messaging JMS Queue

This component provides a common interface to invoke messaging services on WebSphere Default Messaging JMS queues. You can specify the required configuration properties by using the MQSeries component node.

C.1.11.1 WebSphere Default Messaging 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–26, "WebSphere Default Messaging JMS Queue 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–26 WebSphere Default Messaging JMS Queue 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.
Time To Live (seconds)	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. 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.
Provider URL	Enter the provider URL of the JMS implementation used. This is the URL to use for JNDI lookups. <ul style="list-style-type: none"> For WebSphere Default Messaging JMS Queue, set to <code>corbaloc::<DNS Server Name or IP Address>:<bootstrapport></code>
Initial Context Factory	The class name of the initial context factory. This is the starting point for the resolution of names for naming and directory operations. Select WebSphere MQ if you are using WebSphere Default Messaging JMS Queue. This sets the class name to <code>com.ibm.websphere.naming.WsnInitialContextFactory.</code>

Table C–26 WebSphere Default Messaging JMS Queue Sender 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 WebSphere administration console.
Persistent	Select this option if you want the messages to be persistent when dropped into the queue.
Non Persistent	Select this option if you want the messages to be not persistent when dropped into the queue.
Needs compression	Optional. Check this box if the message needs to be compressed before dropping into the queue.
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 Selling and Fulfillment Foundation 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> <p>Note: If you check this box, the system synchronizes the JMS commit along with the DB commit. However, it is not a two-phase commit. Therefore, it may cause some errors, thus resulting in duplicate messages.</p>

Table C–26 WebSphere Default Messaging JMS Queue Sender Configuration Properties

Property	Description
Enable JMS Security	Check this box if you want JMS Security to be enabled. Once selected, the JMS Security Properties tab is enabled to configure Queue and/or JNDI based JMS security.
Header Tab	
Header Name	<p>The name of the message header. For example, APINAME.</p> <p>Note: The header name must be unique. Two headers cannot have the same header name.</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>
Reconnect Tab	
You need to use the settings in this tab for the backup JMS server.	

Table C–26 WebSphere Default Messaging JMS Queue Sender Configuration Properties

Property	Description
Retry Interval (milliseconds)	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.
Number of Retries	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.
Use backup JMS Queue	<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>
Provider URL	<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 WebSphere Default Messaging JMS Queue, set to <code>corbaloc::<DNS Server Name or IP Address>:<bootstrap></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 WebSphere Default Messaging JMS Queue. This sets the class name to</p> <ul style="list-style-type: none"> <code>com.ibm.websphere.naming.WsnInitialContextFactory.</code>

Table C–26 WebSphere Default Messaging JMS Queue Sender Configuration Properties

Property	Description
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 WebSphere administration console.
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 reverts back to using the primary JMS server. If contact with the primary JMS server cannot be established, it continues to use the backup JMS server. The default value is 600 seconds.
<p>JMS Security Properties Tab</p> <p>This is enabled upon selecting Enable JMS Security in the runtime tab.</p> <p>Note: You can override the JMS security properties specified here by defining the agent and flow authorization parameters in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>	
Queue based security	Check this box, if you want to provide Queue based security for JMS service.

Table C–26 WebSphere Default Messaging JMS Queue Sender Configuration Properties

Property	Description
JNDI based security	Check this box, if you want to provide JNDI based security for JMS service.
JMS Security Parameters For more information about application server-specific JMS security parameters, 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 Application 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.

The `yfs.jms.session.reaptime` property in the `yfs.properties` file is used to set the JMS session reaptime. To modify this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_ overrides.properties` file. For additional information about overriding properties using the `customer_ overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

C.1.11.2 WebSphere Default Messaging JMS Queue 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–27 WebSphere Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
Runtime Tab	
Sub Service Name	Enter the name of the queue to which the message is sent.
Queue Name	Enter the name of the queue to which the message is sent.
Provider URL	Enter the JMS server provider URL of the JMS implementation used. This is the URL to use for JNDI lookups. <ul style="list-style-type: none"> For WebSphere Default Messaging JMS Queue, set to <code>corbaloc::<DNS Server Name or IP Address>:<bootstrapport></code>

Table C–27 WebSphere Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
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 WebSphere Default Messaging JMS Queue. This sets the class name to</p> <pre>com.ibm.websphere.naming.WsnInitialContextFactory.</pre>
QCF Lookup	<p>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 WebSphere administration console.</p>
Receiving Mode	<p>Indicates whether the messages are received in a transactional mode or non-transactional mode.</p> <ul style="list-style-type: none"> • If you set this to non-transactional mode, the messages are removed from the queue as soon as it is read. • If you set this to transactional mode, the messages are removed from the queue when they are functionally processed and if an exception is thrown when processing it. <p>If the IsReprocessable flag is checked and an exception occurs when processing the message, then before deleting these messages, ensure that they are added to the YFS_REPROCESS_ERROR or YFS_INBOX table. However, if the "service suspend" exception occurs when adding these deleted messages to a table, the messages are maintained in the queue.</p>

Table C–27 WebSphere Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
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>Selling and Fulfillment Foundation: System Manangement and Administration Guide</i>.</p> <p>You can also start multiple instances of the Service Definition Framework for a specific integration adapter for a specific server. For more information about the integration adapter, see the <i>Selling and Fulfillment Foundation: Performance Management Guide</i>.</p>
Selector	<p>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'.</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 <INSTALL_DIR>/properties/customer_overrides.properties file) before executing this service. For more information see, Section C.1.5.3, "Enabling EOF Messages in the Application Platform Framework". For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>

Table C–27 WebSphere Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
Root Node Name of EOF Message	<p>If the message contains an EOF message ID, enter your custom root node name for the EOF message.</p> <p>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 Application Platform Framework".</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 configure Queue and/or JNDI based JMS security.</p>
Server Tab	
Server Name	<p>Select the name of the server which actually executes the service.</p> <p>For more information about creating a new server, see "Adding a New Server".</p>
Needs Decompression	<p>Select this field to indicate that the incoming message is compressed and needs to be decompressed.</p>
Exception Tabs	<p>See Table C–37 for the receiver link exception handling properties.</p>
Reconnect Tab	
Maximum Time Between Retries (milliseconds)	<p>Enter the maximum time between reconnection attempts.</p>
JMS Security Properties Tab	
<p>This is enabled upon selecting Enable JMS Security in the runtime tab.</p> <p>Note: You can override the JMS security properties specified here by defining the agent and flow authorization parameters in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>	

Table C–27 WebSphere Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
Queue based security	Check this box, if you want to provide Queue based security for JMS service.
JNDI based security	Check this box, if you want to provide JNDI based security for JMS service.
JMS Security Parameters For more information about application server-specific JMS security parameters, 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.

Connection Properties

The following are the WLJMS and MQJMS nodes' connection properties:

Table C–28 WebSphere Default Messaging JMS Queue 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.1.12 JBoss Default Messaging JMS Queue

This component provides a common interface to invoke messaging services on the JBoss Default Messaging JMS queues. You can specify the configuration properties by using the Generic JMS component node.

C.1.12.1 JBoss Default Messaging JMS Queue Sender

The JMS sender caches a single connection starting with the primary queue. If the message is not sent, the JMS sender retries to send the message a configurable number of times with a configurable delay between each attempt as defined for the Reconnect property in [Table C–29, "JBoss Default Messaging JMS Queue Sender Configuration Properties"](#). An exception is thrown if the configured number of retries are exhausted and there is no backup queue.

If a backup queue is set up, a connection is made to the backup queue. The backup queue becomes the current queue for the next attempt to send a message. The sender can be configured to reconnect to the primary queue. If the primary queue is working, 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–29 JBoss Default Messaging JMS Queue Sender Configuration Properties

Property	Description
Runtime Tab	
Use the settings in this tab for the primary JMS queue server.	
Queue Name	Enter the name of the queue to which the message is sent.

Table C–29 JBoss Default Messaging JMS Queue Sender Configuration Properties

Property	Description
Time To Live (seconds)	<p>Enter the time period after which you want the messages to be deleted from the queue.</p> <ul style="list-style-type: none"> • If you enter a zero value, the messages are not deleted from the queue. • If you enter a non-zero value, the messages that are not consumed for the specified time interval are automatically deleted from the queue by the provider. <p>Note: If you do not enter any value, the system considers a zero value.</p>
Provider URL	<p>Enter the provider URL of the JMS implementation to use for JNDI lookups.</p> <p>For the JBoss Default Messaging JMS Queue, enter the URL as: <code>jnp::<DNS Server Name or IP Address>:<bootstrap></code></p>
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 Jboss if you are using the JBoss Default Messaging JMS Queue. This sets the class name to <code>org.jnp.interfaces.NamingContextFactory</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 the one configured in the JBoss administration console.</p>
Delivery Mode	<p>This is not used for JBoss Messaging. Persistence is set for each queue. Do not allow the application to override the persistence.</p>
Needs compression	<p>Optional. Check this box if the message needs to be compressed before dropping it into the queue.</p>

Table C–29 JBoss Default Messaging JMS Queue 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 Selling and Fulfillment Foundation 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 later rolled back from the queue if an error is encountered in the ON_SUCCESS event after the message is staged. However, in the non-transactional mode, the message remains in the queue once it is staged and not rolled back.</p> <p>Note: If you check this box, the system synchronizes the JMS commit along with the DB commit. However, it is not a two-phase commit. Therefore, it may cause some errors, thus resulting in duplicate messages.</p>
Enable JMS Security	Check this box if you want to enable the JMS Security. If you check this box, the JMS Security Properties tab is enabled for you to configure Queue and/or JNDI based JMS security.
Header Tab	

Table C–29 JBoss Default Messaging JMS Queue Sender Configuration Properties

Property	Description
Header Name	<p>The name of the message header. For example, APINAME.</p> <p>Note: The header name must be unique. Two headers cannot have the same header name.</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>This 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>
<p>Reconnect Tab</p> <p>Use the settings in this tab for the backup JMS queue server.</p>	
Retry Interval (milliseconds)	<p>In the event that the connection to the primary JMS queue server is lost, enter the time required to reestablish contact with the primary JMS queue server. This parameter is used in conjunction with the Number of Retries parameter. The default value is 0 (zero), which means no time delay between retry attempts.</p>

Table C–29 JBoss Default Messaging JMS Queue Sender Configuration Properties

Property	Description
Number of Retries	In the event that the connection to the primary JMS queue server is lost, enter the number of attempts to reestablish contact with the primary JMS queue server before failing over to the backup JMS queue server, if enabled, or an exception is thrown. This parameter is used in conjunction with the Retry Interval parameter. The default value is 0 (zero), which means there are no retries if the connection is lost and failover occurs, if enabled, or an exception is thrown.
Use backup JMS Queue	Check this box if you want to enable a backup JMS queue. When you check this box, other controls in this tab are enabled.
Provider URL	Enter the backup JMS queue server provider URL of the JMS implementation used. This is the URL to use for JNDI lookups. For the JBoss Default Messaging JMS Queue, enter the URL as: <code>jnp::<DNS Server Name or IP Address>:<bootstrap></code>
Initial Context Factory	The class name of the initial context factory. This is the starting point for the resolution of names for naming and directory operations. Select Jboss if you are using the JBoss Default Messaging JMS Queue. This sets the class name to <code>org.jnp.interfaces.NamingContextFactory</code>
Queue Name	Enter the name of the backup queue to which the message is sent.
QCF Lookup	Enter the backup queue of the JMS server 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 the one configured in the JBoss administration console.

Table C–29 JBoss Default Messaging JMS Queue Sender Configuration Properties

Property	Description
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 has occurred. Once this box is selected, enter the wait time before it reconnects.
Time Before Reconnect (seconds)	Enter the maximum time required to reestablish contact with the primary JMS queue server after a failover to the backup JMS queue server. If a contact is reestablished, the primary JMS queue server is used. Otherwise, the backup JMS queue server is used. The default value is 600 seconds.
<p>JMS Security Properties Tab</p> <p>This tab is enabled when you select Enable JMS Security option in the runtime tab.</p> <p>Note: You can override the JMS security properties specified here by defining the agent and flow authorization parameters in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i>.</p>	
Queue based security	Check this box, if you want to provide Queue based security for JMS service.
JNDI based security	Check this box, if you want to provide JNDI based security for JMS service. Note: This checkbox is not applicable for JBoss as it does not support JNDI based JMS security.
<p>JMS Security Parameters</p> <p>For more information about application server-specific JMS security parameters, 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 Application 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.

The `yfs.jms.session.reaptime` property in the `yfs.properties` file is used to set the JMS session reaptime. To modify this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

C.1.12.2 JBoss Default Messaging JMS Queue Receiver

The JMS receiver does not have a backup queue. Therefore, you must configure two different services; one to listen on the primary queue and the other to listen on the backup queue. When an exception occurs, the receiver retries to use an exponential back off mechanism with a configurable maximum wait limit between retries.

For example, if 600 seconds is the maximum wait time, the receiver first waits for 1 second, then 2 seconds, 4, 8, 16, and so forth until it reaches 600 seconds, after which, the receiver retries every 600 seconds. Therefore, there is no boundary on the total wait time.

As 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–30 JBoss Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
Runtime Tab	
Sub Service Name	Enter the name of the queue to which the message is sent.
Queue Name	Enter the name of the queue to which the message is sent.
Provider URL	Enter the JMS queue server provider URL of the JMS implementation to use for JNDI lookups. For the JBoss Default Messaging JMS Queue, enter the URL as: <code>jnp::<DNS Server Name or IP Address>:<bootstrapport></code>
Initial Context Factory	The class name of the initial context factory. This is the starting point for the resolution of names for naming and directory operations. Select Jboss, if you are using JBoss Default Messaging JMS Queue. This sets the class name to <code>org.jnp.interfaces.NamingContextFactory</code>
QCF Lookup	Enter the backup queue of the JMS queue server 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 the one configured in the JBoss administration console.

Table C–30 JBoss Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
Receiving Mode	<p>Indicates whether the messages are received in transactional mode or non-transactional mode.</p> <ul style="list-style-type: none"> • If you set this to non-transactional mode, the messages are removed from the queue as soon as they are read. • If you set this to transactional mode, the messages are removed from the queue when they are functionally processed or if an exception is thrown when processing it. <p>If the IsReprocessable flag is checked and an exception occurs when processing the message, before deleting the message, ensure that it is added to the YFS_REPROCESS_ERROR or YFS_INBOX table. However, if the "service suspend" exception occurs when adding deleted messages to a table, the messages are maintained in the queue.</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>Selling and Fulfillment Foundation: System Management and Administration Guide</i>.</p> <p>You can also start multiple instances of the Service Definition Framework for a specific integration adapter and a server. For more information about the integration adapter, see the <i>Selling and Fulfillment Foundation: Performance Management Guide</i>.</p>

Table C–30 JBoss Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
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 the header name='APINAME' and the value='createOrder'.
Service to Execute on EOF Message	Required if the message contains an EOF message ID. Click  to select the service to invoke when an EOF message is received. Once the EOF message is received, the framework waits for a few minutes (configurable in the <INSTALL_DIR>/properties/customer_overrides.properties file) before executing this service. For more information about EOF messages, see Section C.1.5.3, "Enabling EOF Messages in the Application Platform Framework" . For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i> .
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 EOF message has a root node of "EOF". For more information about EOF messages, see Section C.1.5.3, "Enabling EOF Messages in the Application Platform Framework" .
Enable JMS Security	Check this box if you want to enable JMS Security. When this box is selected, the JMS Security Properties tab is enabled for you to configure Queue and/or JNDI based JMS security.
Server Tab	

Table C–30 JBoss Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
Server Name	Select the name of the server that executes the service. For more information about creating a new server, see "Adding a New Server" .
Needs Decompression	Check this box to indicate that the incoming message is compressed and needs to be decompressed.
Exception Tabs	See Table C–37 for the receiver link exception handling properties.
Reconnect Tab	
Maximum Time Between Retries (milliseconds)	Enter the maximum time required between reconnection attempts.
JMS Security Properties Tab	
This tab is enabled only when you check the Enable JMS Security box provided in the runtime tab. Note: You can override the JMS security properties specified here by defining the agent and flow authorization parameters in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i> .	
Queue based security	Check this box, if you want to provide Queue based security for JMS service.
JNDI based security	Check this box, if you want to provide JNDI based security for JMS service. Note: This checkbox is not applicable for JBoss as it does not support JNDI based JMS security.
JMS Security Parameters	
For more information about application server-specific JMS security parameters, see "Setting up the JMS Security Properties" section.	

Table C–30 JBoss Default Messaging JMS Queue Receiver Configuration Properties

Property	Description
Parameter Name	Enter the name of the security parameter.
Parameter Value	Enter the value of the security parameter.

Connection Properties

The following are the Generic JMS queue node connection properties:

Table C–31 JBoss Default Messaging JMS Queue 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.1.13 Synchronous WebSphere Default Messaging

The synchronous WebSphere Default Messaging 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 in order for it to be picked up and processed correctly.

Configuration Properties

The following are the properties of this node:

Table C–32 Synchronous WAS Default Messaging Properties

Property	Description
Runtime Tab	
Provider URL	Enter the provider URL of the JMS implementation used. URL to use for JNDI lookups using the WebSphere Default Messaging JMS queue: <code>corbaloc::<DNS Server Name or IP Address>:<bootstrapport></code>
Initial Context Factory	The class name of the initial context factory. This is the starting point for the resolution of names for naming and directory operations. Select synchronous WebSphere MQ if you are using MQSeries accessed through a WebSphere corbaloc URL. This sets the class name to <code>corbaloc::<DNS Server Name or IP Address>:<bootstrapport></code>
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 unique identifier for the QCF lookup. This name must be the same as that configured in the WAS administration console.
Needs Compression	Select this option if the message needs to be compressed.
Enable JMS Security	Check this box if you want JMS Security to be enabled. Once checked, the JMS Security Properties tab is enabled to configure Queue and/or JNDI based JMS security.
Request Tab	
Queue Name	Enter the name of the queue to which the request message is sent.

Table C–32 Synchronous WAS Default Messaging Properties

Property	Description
Header Name	<p>The name of the message header. For example, APINAME.</p> <p>Note: The header name must be unique. Two headers cannot have the same header name.</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>
Response Tab	
Queue Name	Enter the name of the queue in which the response message is received.
Selector	Enter the selectors based on the message headers. When specifying a selector, use only single quotation marks. For example, specifying the selector APINAME='createOrder' selects all messages with a header name='APINAME' and value='createOrder'.
Time Out (seconds)	This is a mandatory parameter. Enter the time interval (in seconds) after which the request times out.
Reconnect Tab	
Retry Interval (milliseconds)	In the event that the connection to the JMS server is 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 zero, implying no delay time between retry attempts.

Table C–32 Synchronous WAS Default Messaging Properties

Property	Description
Number of Retries	In the event that the connection to the JMS server is 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 zero, implying no retries if the connection is lost and an exception is thrown immediately.
JMS Security Properties Tab This is enabled when you select Enable JMS Security in the Runtime Tab. Note: You can override the JMS security properties specified here by defining the agent and flow authorization parameters in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i> .	
Queue based security	Check this box, if you want to provide Queue based security for JMS service.
JNDI based security	Check this box, if you want to provide JNDI based security for JMS service.
JMS Security Parameters. For more information about application server-specific JMS security parameters, see " Setting up the JMS Security Properties ".	
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 Application 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.

The `yfs.jms.session.reaptime` property in the `yfs.properties` file is used to set the JMS session reaptime. To modify this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_ overrides.properties` file. For additional information about overriding properties using the `customer_ overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

Connection Properties

The following are the synchronous MQSeries nodes' connection properties:

Table C–33 Synchronous WAS Default Messaging 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.1.14 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.1.14.1 MSMQ Sender

Configuration Properties

The following are the properties of this node:

Table C–34 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> Selling and Fulfillment Foundation 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.

Table C-34 MSMQ Sender Configuration Properties

Property	Description
Transactional/Non Transactional	<p>Select Transactional if you want the message to be committed to the queue only after the service is completed.</p> <p>Select Non Transactional if you want the message to be committed to the queue immediately.</p> <p>For example, if the ON_SUCCESS event of any standard Selling and Fulfillment Foundation 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> <p>Note: If you plan to use transactional mode of messaging, create the queue in MSMQ as transactional queue.</p>
String Message	<p>Select this field if you want to set the PROPID_M_BODY_TYPE to VT_BSTR</p> <p>De-select this field if you want to set the PROPID_M_BODY_TYPE to VT_EMPTY.</p>

Note: The JMS session objects can be pooled based on the service being executed. Hence, whenever the JMS sender requires a session object, the Application 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.

The `yfs.jms.session.reaptime` property in the `yfs.properties` file is used to set the JMS session reaptime. To modify this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

C.1.14.2 MSMQ Receiver

Configuration Properties

The following are the properties of this node:

Table C–35 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> Selling and Fulfillment Foundation does not support Queue Lookup through the Active Directory.
Max Threads	The maximum number of threads for this process.

Table C–35 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 about creating a new server, see "Adding a New Server" .
Exception Tabs	See Table C–37 for the receiver link exception handling properties.

Connection Properties

The following are the MSMQ node's connection properties:

Table C–36 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 <INSTALL_DIR>/bin directory and the directory containing the jvm.dll are in your system PATH.

C.1.15 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, MQSeries, and TIBCO JMS Receiver](#)
- [MSMQ Receiver](#)

Configuration Properties

The following are the exception properties of the asynchronous receiver nodes:

Table C–37 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 displays 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>Selling and Fulfillment Foundation: Javadocs</i> .

Table C–37 Receiver Link Exception Handling Properties

Property	Description
Suspend Wait Time	Enter the time to wait before attempting to reprocess the message.
Is Reprocessible	<p>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.</p> <p>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>Selling and Fulfillment Foundation: Application Platform User Guide</i>.</p> <p>Note: There will only be one reprocess-timer thread per integration server. The reprocess-timer thread will run with a sleep delay of ONE minute and will process 50 reprocessible records at a time.</p>
Check for Prior Exception	<p>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.</p> <p>Note: This option is applicable only when all the related services are associated to the same server.</p>
Exception Group	<p>Enter a group of related services where exceptions are linked. For example, two services for receiving modifications on an order from external systems.</p> <p>Note: This option is applicable only when all the related services are associated to the same server.</p>
Prior Errors User Exit	<p>Enter the class name that implements <code>YIFErrorSequenceUE</code> user exit for checking prior related errors for the message.</p> <p>For more information about <code>YIFErrorSequenceUE</code>, see the <i>Selling and Fulfillment Foundation: Javadocs</i>.</p>

Table C–37 Receiver Link Exception Handling Properties

Property	Description
Exception References Tab	
Exception Reference Name	Enter the name of the exception reference. Saved in the ERROR_REFERENCE column of the YFS_REPROCESS_ERROR table to indicate as Name=Value. For example, NAME1.
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 xml://<full path of the element from root>/@<attribute name>.</p> <p>For example, to get the sales order number from the createOrder input XML, use xml://Order/@OrderNo. 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–38 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 displays 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.
Description	Enter a brief description of the alert raised.

Table C–38 Alert Configuration Properties

Property	Description
Template Name	<p>Enter the XSL Template Name.</p> <p>The output of the XSL and the incoming XML document merge displays in the description field on the Alert Console. For example, to display OrderNo, the sample XSL is</p> <pre><INSTALL_ DIR>/repository/xapi/template/merged/e xception_console/sample_exception_ console.xsl</pre> <p>When the type of incoming XML is TYPE_ JAVA_MAP, to display the OrderNo, the sample XSL is <INSTALL_ DIR>/repository/xapi/template/merged/e xception_console/sample_keymap_ exception_console.xsl.</p> <p>This XSL is called by the raiseEvent API. For more information about the incoming XML data types, see the <i>Selling and Fulfillment Foundation: Javadocs</i>.</p>
List Template	Enter the XSL list template. The output of the XSL and the incoming XML document merge displays 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.
System Arguments Tab	
Argument Name	<p>Predefined elements available to a particular process repository. The following attributes are optional and can be specified as static or dynamic values.</p> <pre>xml:/<ElementName>/@<AttributeName></pre>

Table C–38 Alert Configuration Properties

Property	Description
Argument Value	Static or dynamic values that are specified in the following format: xml: /<ElementName>/@<AttributeName>
User Arguments Tab	
Reference Name	Enter additional name/value data that can be used to add information to the alert.
Reference Value	Enter the reference value. This is a static or dynamic values that are specified in the following format: xml: /<ElementName>/@<AttributeName>
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–38 Alert Configuration Properties

Property	Description
Consolidation Template	<p>Select either <code>Default</code> or <code>Custom</code> for consolidating the alerts. The default chosen option is <code>Default</code>.</p> <p>If <code>Custom</code> is chosen, then the attributes for the consolidation template appear below. These attributes are from the <code>YFS_INBOX</code> table and the corresponding XML file can be found in <code><INSTALL_DIR>/repository/xapi/template/merged/resource/exceptionConsolidation.xml</code>. 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 <code>YFS_INBOX</code> table, then those attributes appear under the <code>Extn</code> element.</p>
Consolidate Dates By	<p>Specify whether the consolidation is done daily or hourly. By default the option is set to <code>Day</code>.</p> <p>Choosing this option is useful only if you include any time attributes such as <code>Date</code> in the consolidation template.</p>
Routing Tab	
Routing Required	<p>Check this box, if you want to route the alert based on the routing rules. For more information about defining routing rules, see Section 17.3.3, "Creating an Exception Routing Rule".</p>
Routing Organization	<p>Enter the XML path of the organization that owns the alert routing rules.</p> <p>By default, this field is disabled.</p>

Table C–38 Alert Configuration Properties

Property	Description
Attribute Name	<p>Displays the predefined routing attributes applicable for an alert. You can specify the following predefined routing attributes:</p> <ul style="list-style-type: none"> • BUYER • SELLER • ENTERPRISE • NODE • DEPARTMENT CODE <p>Note: You can also enter your own custom routing attributes.</p>
Attribute XPath	Enter the XML path of the routing attribute.

Connection Properties

The following are the alert node's connection properties:

Table C–39 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 Selling and Fulfillment Foundation System APIs or any user-written Java class.

Note: To configure extended database APIs for custom and hang-off entities, refer to the *Selling and Fulfillment Foundation: Extending the Database Guide*.

Configuration Properties

The following are the properties of this node:

Table C–40 API Configuration Properties

Property	Description
General Tab	
Standard Selling and Fulfillment Foundation API	Select this option if a standard Selling and Fulfillment Foundation API is to be invoked. If selected, a Standard Selling and Fulfillment Foundation API Name drop down list displays. 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.

Table C-40 API Configuration Properties

Property	Description
Extended Database API	<p>Select this option if the service invokes a custom or hang-off API. If selected, a custom API Name drop-down list displays. For each API, the Class Name and Method Name are provided and cannot be edited.</p> <p>Note: If you want to lock a record in a custom table, pass the <code>SelectMethod</code> attribute as part of the input XML to the custom entity API. The <code>SelectMethod</code> attribute can take the following values: WAIT, NO_WAIT, and NONE.</p> <p>For more information about locking records in Extended APIs, refer to the <i>Selling and Fulfillment Foundation: Customizing APIs Guide</i></p>
API Name	<p>Select or enter the API to be called.</p> <p>Note: This field is for integration purposes only.</p>
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 Selling and Fulfillment Foundation system APIs).
Version	If you chose Requires Backward Compatibility, select the Selling and Fulfillment Foundation version the API is to behave as. Only the applicable versions for the individual API display.
Arguments Tab	

Table C-40 API Configuration Properties

Property	Description
Argument Name	<p>You can pass name/value pairs to the API by entering the values in the Arguments Tab.</p> <p>In order for custom APIs to access custom values, the API should implement the interface <code>com.yantra.interop.japi.YIFCustomApi</code>.</p> <p>If entered, these name/value pairs are passed to the CustomApi as a Properties object.</p>
Argument Value	Enter the argument value.
Template Tab	
	<p>When the Selling and Fulfillment Foundation 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 Hierarchy tree, or both. However, if the template has been specified at both definition levels, the template specified in the Service Definition is used.</p>
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.
Facts Tab	
	<p>A Fact is an attribute that is used by an API or an agent in Selling and Fulfillment Foundation to identify which colony to connect to and retrieve data from. Based on the fact name and fact value entered, the corresponding colony is determined.</p>

Table C–40 API Configuration Properties

Property	Description
Fact Name	Enter the fact name of the XML attribute.
Fact Value	Enter the fact value of the XML attribute.

Connection Properties

The following are the API node's connection properties:

Table C–41 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
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–42 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–43 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–44 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 for 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–45 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-46 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.
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. Selling and Fulfillment Foundation supplies the <code><INSTALL_DIR>/repository/xapi/template/merged/email/orders_mail.xsl</code> file.

Table C–46 E-Mail Configuration Properties

Property	Description
From	Can be static or dynamic with the XML path specified as xml://<ElementName>/@<AttributeName>. Use semi-colons as a delimiter between addresses.
To	Required. Can be static or dynamic with the XML path specified as xml://<ElementName>/@<AttributeName>. Use semi-colons as a delimiter between addresses.
CC	Can be static or dynamic with the XML path specified as xml://<ElementName>/@<AttributeName>. Use semi-colons as a delimiter between addresses.
BCC	Can be static or dynamic with the XML path specified as xml://<ElementName>/@<AttributeName>. Use semi-colons as a delimiter between addresses.

Connection Properties

These are the connection properties of this node:

Table C–47 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–47 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 Application 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 about setting the properties for connection reaptime, check the `yfs.properties` file in `<INSTALL_DIR>/properties` directory.

The `yfs.smtp.session.reaptime` property in the `yfs.properties` file is used to set the JMS session reaptime. To modify this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

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–48 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–49 Nomenclature Runtime Configuration Properties

Property	Description
XML Name	Select XML name created through nomenclature configuration. For more information about nomenclature configuration, see Section 11.3, "Defining Nomenclature Configuration" .

Connection Properties

The following are the Nomenclature Runtime connection properties of this node:

Table C–50 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

Table C–50 Nomenclature Runtime 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	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–51 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	<p>Enter the XML path where the participant's organization code is located. For example, xml://Order/@OrganizationCode.</p> <p>Participant preferences are appended to the document before executing the specified service.</p>

Connection Properties

The following are the Router node's connection properties:

Table C–52 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 Selling and Fulfillment Foundation. 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–53 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.

Table C-53 Text Translator Configuration Properties

Property	Description
Input Text Format	<p>If you selected Input Data Format to be Text select:</p> <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	<p>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.</p> <p>For example, if the flat file schema XML file is located in <INSTALL_DIR>/bin/test.xml, and <INSTALL_DIR> is in the CLASSPATH, then the value of the attribute is /bin/test.xml.</p>

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 must be text
- 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–54 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 *Selling and Fulfillment Foundation*, a classpath can be used to find the XSL files included by the `xsl:include` directive. You can define the class to provide a custom `URIResolver` during XSL processing using the `yfs.xsl.uriresolver` property. By default, the value of this property is set to `com.yantra.interop.util.YantraDefaultURIResolver`.

If you want to specify a different class name, modify the `yfs.xsl.uriresolver` property in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

Note: If the `yfs.xml.uriresolver` property is set to another class name, this new class will be used in place of the default `YantraDefaultURIResolver` class. The new class must implement the `javax.xml.transform.URIResolver` interface.

Configuration Properties

The following are the properties of this node:

Table C-55 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-56 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-57](#) are optional.

Table C-57 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 data-bbox="751 458 1315 626"> <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>YIFXXMLAttrOverride</code> interface. For more information about this interface, see the <i>Selling and Fulfillment Foundation: 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–57 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	<p>Select this option if you want to use a static attribute value.</p> <p>Once this option is selected you can enter the value and check the override for an existing value.</p>

Table C–57 Defaulting Component Configuration Properties

Property	Description
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 user groups discussed in [Section 5.2, "Defining User 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 about the `YIFSecurityValidator`, see the *Selling and Fulfillment Foundation: 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–58 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.				
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 1003 1268 1190"> <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. Additionally, you can also generate the PDF object of a document. 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 be printed conveniently by a single click from the console. This is supported by using flow execution and the printer component.

Jasper Print component can also be used to generate a PDF report and stream it down via HTTP response. This can be done by configuring a service with a Jasper Print component and selecting the "Send as PDF stream" option. This service is now invoked by making a HTTP call to the "InteropJasperServlet" servlet and passing the input XML "locale" to the service and the name of the service that you created for generating PDF objects. These services are created in the Service Definition Framework (SDF). For more information about SDF and how to create service, see [Chapter 4, "Configuring Process Models"](#). The PDF that is generated is streamed back by the servlet as a PDF stream.

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–59 Jasper Printing Configuration Properties

Field	Description
General Tab	
Jasper Report	This is a compiled Jasper file. The file name supports variables in both a standard and enhanced variable format as explained in the <i>Selling and Fulfillment Foundation: Customization Basics 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.
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" .
Use Selling and Fulfillment Foundation Resource Bundle	If checked, all literals in the report are translated to the requested locale using the Selling and Fulfillment Foundation Resource Bundle.
Output Tab	
Printer	Select this option to specify a printer to use.
Printer Name	Enter the XML path pointing to the printer to use.
Number of Copies	Enter the number of copies you wish to print.
PDF Byte Stream	Select this option to generate a PDF object.
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 g xml:/Order/@OrganizationCode
```

```
doctype g 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`. To modify this property, add an entry for it in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

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 Service Definition Framework provides hooks to connect to external systems like Sterling GIS through custom adapters. Selling and Fulfillment Foundation currently provides the following standard adapter:

- [Sterling GIS](#)

C.3.1 Sterling GIS

This adapter is used for communicating with the Sterling Gentran Integration Suite 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 about the fix pack appropriate for your JDK version, see the *Selling and Fulfillment Foundation: Installation Guide*.

Configuration Properties

The following are the properties of this adapter:

Table C–60 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).
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 the <code><INSTALL_DIR>/properties/customer_overrides.properties</code> file. For additional information about overriding properties using the <code>customer_overrides.properties</code> file, see the <i>Selling and Fulfillment Foundation: Properties Guide</i> .
Key Store Passwd	If Is Secure is selected, enter the password to access the key store.

Table C–60 Sterling GIS Configuration Properties

Property	Description
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 the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the <i>Selling and Fulfillment Foundation: Properties 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–61](#):

Table C–61 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.

D

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 displays 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 displays 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 Selling and Fulfillment Foundation 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 `<INSTALL_DIR>/xapidocs/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 Selling and Fulfillment Foundation 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. Selling and Fulfillment Foundation supplies four XSDs—two XSDs for data coming in from positional and delimited files and two XSDs for data from Selling and Fulfillment Foundation to convert to positional and delimited flat files.

These files follow the standards defined by the World Wide Web Consortium (W3C); for information about this standard, see <http://www.w3.org/XML/Schema>.

All four XSD files are designed specifically to correspond with the Selling and Fulfillment Foundation 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 Selling and Fulfillment Foundation.

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 `<INSTALL_DIR>/xapidocs/code_examples/flatfile/` directory.

Note: Sterling Commerce recommends that you periodically check for files within the `/error/` and `/complete/` directories that contain errors or are 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 `<INSTALL_DIR>/xapidocs/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 are 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 flat file.
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 order line, and an order line could contain a sequence of one or more line items. In this case, the order line 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 flat file.
<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 displays 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
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 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 is 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	
<code>RecordIdStartPosition</code>	Required. Integer. Indicates the start position of a record.
<code>DefaultRecordDelimiter</code>	Required. <code>RecordDelimiterType</code> . The default delimiter between records is <code>Newline</code> .
<code>DefaultFieldDelimiter</code>	Optional. <code>CharacterType</code> . The default field delimiter between the fields is comma <code>[,]</code> .
<code>DefaultEscapeCharacter</code>	Optional. <code>CharacterType</code> . 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 are 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="656 807 1320 1310"><?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 is set to _@_.</p> <p>Note: The default wrap character attribute is optional, and by default has a value of "". If DefaultWrapCharacter="NONE", 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 <INSTALL_DIR>/bin/test.xml file, and <INSTALL_DIR> 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.	
Recorded	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.
Smiling	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.
Smiling	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
Field Position	Required. Integer The position of this field within the record. If the Recorded is at position 1, then the numbering of fields should begin at position 2.
Containment Type	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 is created with the tag name set to Smiling 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
Parser Defaults Element	
Record Id Start Position	Required. Integer. This field indicates the start position of the Recorded for each record.
Record Id End Position	Required. Integer. This field indicates the end position of the Recorded for each record.
Default Record Delimiter	Required. Record Delimiter Type. The default delimiter between records is <code>Newline</code> .
Default Escape Character	Optional. Characterizes. The default escape character is <code>\.</code>

Table D-4 Elements in the Positional Flat File Sender XSD File

Property	Description
Default Pad Character	<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 are 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 ContainmentType.	
Name	Required. NMTOKEN. The name of the field which is used as a reference in the AttributeMap. 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 are 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 `<INSTALL_DIR>/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
-Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp.DocumentBuilderFactoryImpl"
```

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.DocumentBuilderFactoryImpl com.yantra.integration.adapter.IntegrationAdapter TextTranslatorJMS
```

E

Document Types

This appendix provides a reference for the different document types used by Selling and Fulfillment Foundation.

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 takes 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 warehouse, a distribution center to a store).

Table E-1 Document Types

Document Type	Number	Description
Master Order	0007	This document type is used to sell items that ship in a timed sequence, either from business to business or business to customer.
Load		
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.

Table E-1 Document Types

Document Type	Number	Description
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.
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	This document type is used in two different contexts: <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 Applications Manager. 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](#)
- [Sales Order Receipt](#)

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

Master Order

- Master Order Fulfillment

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](#)

Item-Based Allocation (IBA)

- [Item-Based Allocation \(IBA\) Order](#)

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.

Table F-1 Order Fulfillment Condition Builder Attributes

Attribute	Description
Line Type	The type of the order line. Selling and Fulfillment Foundation 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. Selling and Fulfillment Foundation 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 run 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.
Enterprise Code	The code of the enterprise on the order.
Receiving Node	The node that receives 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 ships the shipment for the order.
Ship Node Interface Type	The interface type of the ship node on the order (External Application, Console, Sterling WMS, or 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.

Table F–1 Order Fulfillment Condition Builder Attributes

Attribute	Description
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.
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 is set to EXCHANGE.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

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.
Payment Terms	The payment terms for this order.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

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 ships this shipment.
Ship Node Interface Type	The interface type of the ship node on the order (External Application, Console, Sterling WMS, or WMS 6.2).
Receiving Node	The node that receives this shipment.
Ship Mode	The shipment mode that is 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.
Routing Guide Maintained	The flag indicating whether a routing guide is maintained for this shipment.
Carrier	The carrier for the shipment.

Table F–3 Outbound Shipment Condition Builder Attributes

Attribute	Description
Real-time Integration with WMS 6.2	The flag indicating whether the node this shipment is shipping from is integrating with the Sterling WMS. Setting this field to N means that you are integrating with WMS 6.2, or any other warehouse management system.
Manually Entered	The flag indicating whether or not the shipment was entered through the Console.
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.
Shipment Type	Indicates a set of shipments that are of the same nature.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i>.</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

F.1.4 Sales Order Receipt

The Sales Order Receipt condition builder attributes are identical to the [Return Receipt](#) attributes.

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. Selling and Fulfillment Foundation has no application logic associated with the return line type. This field can be set up as per your business practices.

Table F–4 Return Fulfillment Condition Builder Attributes

Attribute	Description
Order Type	The type of the return. Selling and Fulfillment Foundation 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 run 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.
Enterprise Code	The code of the enterprise on the return.
Receiving Node	The node that receives 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 be ships the shipment for the return.
Ship Node Interface Type	The interface type of the ship node on the return (External Application, Console, Sterling WMS, or 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.

Table F–4 Return Fulfillment Condition Builder Attributes

Attribute	Description
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.
Is Derived Line	The flag indicating whether the return line is derived from another return line.
Order Purpose	This field is only applicable to sales orders.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

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, Console, Sterling WMS, or WMS 6.2).
Ship Mode	The shipment mode that is used for the shipment. For example, Parcel, Truck Load, Less-Than Truck Load.
Freight Terms	The freight terms on the receipt.
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.

Table F–5 Return Receipt Condition Builder Attributes

Attribute	Description
Is Receiving Node Integrated Real Time	The flag indicating whether the receiving node is integrating with WMS 6.2, or with another WMS system.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

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. Selling and Fulfillment Foundation 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. Selling and Fulfillment Foundation 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 run 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 receives 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 ships the shipment for the inbound order.
Ship Node Interface Type	The interface type of the ship node on the inbound order (External Application, Console, Sterling WMS, or WMS 6.2).

Table F-6 Purchase Order Execution Condition Builder Attributes

Attribute	Description
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	
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.

Table F–6 Purchase Order Execution Condition Builder Attributes

Attribute	Description
Order Purpose	This field is only applicable to sales orders.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i>.</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation 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.

Table F–7 Purchase Order Negotiation Condition Builder Attributes

Attribute	Description
Payment Terms	The payment terms for this inbound order.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation 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 Master Order

F.7.1 Master Order Fulfillment

Table F–8 Master Order Fulfillment Condition Builder Attributes

Attribute	Description
Master 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. Selling and Fulfillment Foundation 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. Selling and Fulfillment Foundation 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 run on the order.

Table F–8 Master Order Fulfillment Condition Builder Attributes

Attribute	Description
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.
Enterprise Code	The code of the enterprise on the order.
Receiving Node	The node that receives 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 ships the shipment for the order.
Ship Node Interface Type	The interface type of the ship node on the order (External Application, Console, Sterling WMS, or 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.

Table F–8 Master Order Fulfillment Condition Builder Attributes

Attribute	Description
Reservation Mandatory	The flag indicating whether the reservation is mandatory.
Related Master 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.
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 is set to EXCHANGE.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

F.8 Load Execution

Table F–9 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.

Table F–9 Load Execution Condition Builder Attributes

Attribute	Description
Carrier Service Code	The code of the carrier service used to carry the load.
Ship Mode	The shipment mode that is 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.
Multiple Load Stop	The flag indicating whether or not a shipment goes through multiple stops to load or unload additional shipments.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

F.9 General

Table F–10 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 ships this shipment.
Supply Type	The supply type associated with the inventory status. Typical values are Onhand, Held, etc.

Table F-10 General Condition Builder Attributes

Attribute	Description
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.
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.
SC UI Client Version	The Rich Client Platform application version number.

Table F–10 General Condition Builder Attributes

Attribute	Description
Activity Group ID	The identifier for the activity group.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

F.10 WMS Putaway

The WMS Putaway condition builder attributes are identical to the [General](#) attributes.

F.11 WMS Layout Definition

The WMS Layout Definition condition builder attributes are identical to the [General](#) attributes.

F.12 WMS Inventory

The WMS Layout Inventory condition builder attributes are identical to the [General](#) attributes.

F.13 Trailer Loading

The Trailer Loading condition builder attributes are identical to the [General](#) attributes.

F.14 Task Execution

The Task Execution condition builder attributes are identical to the [General](#) attributes.

F.15 Move Request Execution

The Move Request Execution condition builder attributes are identical to the [General](#) attributes.

F.16 Manifesting

The Manifesting condition builder attributes are identical to the [General](#) attributes.

F.17 Over Pack Build

The Over Pack Build condition builder attributes are identical to the [General](#) attributes.

F.18 Count Execution

Table F–11 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 are performed at the LPN level.
Is Case Level	The flag indicating whether the count tasks are performed at the case level.
Is Pallet Level	The flag indicating whether the count tasks are performed at the pallet level.

Table F–11 Count Execution Condition Builder Attributes

Attribute	Description
Is Item Level	The flag indicating whether the count tasks are 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–11 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 displays.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i>.</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

F.19 Pack Process

Table F–12 Pack Process Condition Builder Attributes

Attribute	Description
Node Attributes	
Ship Node	The node that ships this shipment.
Receiving Node	The node that receives this shipment.
Ship from Ship Node Interface Type	The interface type of the ship node from which the shipment is shipped (External Application, Console, Sterling WMS, or 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 WMS 6.2.
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, Console, Sterling WMS, or WMS 6.2).

Table F–12 Pack Process Condition Builder Attributes

Attribute	Description
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 WMS 6.2.
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 is 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.

Table F–12 Pack Process Condition Builder Attributes

Attribute	Description
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.
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 about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i>.</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

F.20 Outbound Picking

Table F–13 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 about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

F.21 VAS Process

Table F–14 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 run.
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.

Table F–14 VAS Process Condition Builder Attributes

Attribute	Description
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 is 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 run 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 run the service.
Last Execution Success	The flag indicating whether the last attempt to run the service was successful or not.
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 about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i> .</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

F.22 Item-Based Allocation (IBA) Order

Table F-15 IBA Attributes

Attribute	Description
Order Attributes	
Exchange Type	The exchange type of the order.
Priority Code	Customizable priority code of the order.
Priority Number	The numeric priority code of the order.
Document Type	The document type for this order. Typical value is 0001 (Sales Order).
Order Type	The order classification attribute. This field can be used for reporting purposes or to build conditions for modeling your business process. Selling and Fulfillment Foundation has no default logic based on this field.
Entry Type	The channel through which this order was created.
Department Code	Department code to which the order was placed.
Search Criteria 1	Customizable field for allowing searches.
Search Criteria 2	Customizable field for allowing searches.
Order Line	
Line Type	The line type can be used in process modeling for pipeline determination or conditional processing.
Condition Variable 1	A user-defined variable that can be used for condition building in process modeling.
Condition Variable 2	A user-defined variable that can be used for condition building in process modeling.
Shipping Attributes	
Level of Service	The order or the line's level of service.

Table F-15 IBA Attributes

Attribute	Description
Ship To ID	The ship-to identifier. If a customer definition representing the buyer organization exists within Selling and Fulfillment Foundation, the ship-to ID can represent the Customer ID. Otherwise, the ship-to ID can represent the PersonID of the ship-to address or the receiving node of the order.
Carrier Service Code	The code of the carrier service used to carry the load.
Participant Attributes	
Enterprise Code	The code of the enterprise on the order.
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.
Bill To ID	The identifier of the customer to whom the order is being billed.
{Enter Your Own Attribute}	<p>A customizable condition builder attribute. For more information about customizing this field, see the <i>Selling and Fulfillment Foundation: Extending the Condition Builder Guide</i>.</p> <p>Note: This field is limited only to unexposed key attributes that are pre-defined by Selling and Fulfillment Foundation as opposed to any XML attribute that you can enter.</p>

A

- Abort Processing on Error field, 419
- access policies, 287
- Account Number With Hub field, 39, 152
- Action Code field, 245
- Action Group field, 245
- Action Name field, 100, 245
- actions, 243
 - creating, 244
 - deleting, 249
 - modifying, 249
- Active field, 264
- adapter nodes, 250, 252
- Administered By field, 43
- Advanced XML Condition
 - Advanced XML field, 240
 - Condition Cases field, 241
 - Condition Group field, 240
 - Condition ID field, 240
 - Condition Name field, 240
 - Source View field, 241
- AdvancedXML field, 237
- Agent Criteria Group field, 98
- Agent Server field, 203
- Alert Console template, 247
- alert nodes, 850
- Alert Queue Name field, 203, 847
- alert queues, 421
 - based on size, 423
 - creating, 421
 - deleting, 427
 - modifying, 426
 - unassigned alerts, 424
 - unresolved alerts, 425
 - user list, 426
- Alert Type field, 847
- alerts, 55, 60, 421
- Allow Chained Order Creation field, 179
- Allow Demand Updates For The Seller Organization field, 177
- Allow Inventory Check During Schedule And Release field, 177
- Allow Invoice Creation field, 178
- Allow Other Inventory Operations While Physical Count Is On, 126, 130
- Allow Payment Processing field, 178
- Allow Payment Processing flag, 178
- Allow Price Calculation For Confirmed Orders field, 179
- Allow Price Calculation For Draft Orders field, 178
- Allow Pro Forma Invoice Creation For Shipments flag, 178
- Allow Propagation Of Changes To Derived Parent field, 180
- Allow Shipping Of Hazardous Materials field, 87, 91, 95
- Allow Supply Updates For The Buyer Organization field, 177
- API Name field, 386
- API nodes, 855
- Application menu
 - modifying, 370
- application rules side panel, 12
- application versions
 - creating, 327
 - deleting, 328
 - modifying, 328

Applications Manager

- actions, 28
 - document types, 29
 - entering dates/times, 34
 - lists, 31
 - lookup functionality, 28
 - on-line help, 34
 - special characters, 35
 - troubleshooting, 34
 - users, 31
- layout, 10
- starting, 9
- work area, 23
- applicationversionandcode, 327
- Apply SCM on all cases field, 78
- Apply SCM on all shipping units field, 78
- Appointment Required field, 63
- asynchronous transport nodes, 251
- Attribute Location field, 418
- Audit Queue field, 423
- Authority Type field, 152
- Available field, 264, 292

B

- Backup Protocol field, 143, 146, 149
- backwards compatibility
 - order hold deprecated functionality, 316
- bar codes, 312
- Base Currency field, 362
- base document types, 168
- base repositories, 183
- Base Transaction Name field, 199
- base transactions, 195
- Batch Sheet for picking, 455
- Billing Address field, 264
- Binding field, 390
- BOL Prefix field, 98
- Break Bulk Node field, 73
- building
 - catalog index, 532
- business communication components, 6, 403
 - business document codes, 407
 - document format codes, 405
 - protocol codes, 403

- business document codes
 - creating, 407
 - deleting, 408
 - modifying, 408
- business documents
 - classifying, 408
- business models, 2
- buyer documents, 142, 144, 145, 409
- Buyer Documents tab, 142
- buyers, 3, 46, 148
 - configuring data access, 290
 - packaging parameters, 77
 - print format preferences, 83

C

- Calendar ID field, 156
- calendar inheritance, 154
 - child calendar, 154
 - limitations, 155
 - parent calendar, 154
- calendars
 - creating an exception, 159
 - defining an organization's, 154
 - defining defaults, 158
 - inheritance See also calendar inheritance, 154
- Call DB Extension field, 248
- Call HTTP Extension field, 248
- Call Java Extension field, 247
- Can Enterprise Configure Event Handler field, 214
- capacity consolidation level, 314
- Capacity Organization Is field, 135
- capacity organizations, 134
- CAPACITYPRG purge rule, 302
- carrier documents, 148, 150, 151, 409
- Carrier Documents tab, 148
- Carrier field, 115
- Carrier Service Code field, 73
- carrier services, 84
- carriers, 46, 148
 - defining attributes, 84
 - defining less-than truckload carrier services, 88
 - defining parcel carrier services, 92
 - defining truckload carrier services, 85

- roles
 - carriers, 46
- Carrier/Service field, 73, 90
- Cart Manifest for picking, 455
- catalog
 - index building, 532
- Catalog Defined By field, 133
- catalog model, 314
- catalog organization, 133
- catalogs, 133
- Chained Document Type field, 199
- chained orders, 108, 179
- Chained Procurement Purchase Order Document Type field, 179
- Chained Procurement Transfer Order Document Type field, 179
- Check Digit Determination field, 116
- Check for Prior Exception field, 848
- child organizations
 - viewing, 153
- Class Name field, 238, 247, 414
- Client Date Time Format field, 368
- Client Time Format field, 368
- Close Window On Complete field, 391
- Code Description field, 413
- Code Name field, 414
- Code Value API field, 414
- common codes, 396
- communication, 6
- communication protocols, 140
 - creating, 140
 - deleting, 142
 - modifying, 141
- Compliance Service field, 82
- compliance services, 79
 - creating, 80
 - deleting, 83
 - modifying, 83
- component nodes, 250, 252, 849
- composite service nodes, 858
- condition builder, 909 to 937
 - using, 238
- Condition Group field, 237
- condition groups, 237, 240
- Condition ID field, 237
- Condition Name field, 237
- condition nodes, 859
- Condition Properties field, 238
- Condition Value field, 237
- conditions, 234
 - creating, 236
 - deleting, 241
 - modifying, 239
 - viewing affected entities, 241
- configuration screens
 - accessing, 13
- configuration version labels
 - creating, 478
 - deleting, 479
 - modifying, 479
- Configuring, 37
- Connector node, 250
- console actions, 389
- console APIs, 385
- console entities, 378
- console icons, 392
- console inner panels, 388
- console links, 391
- console list views, 384
- console related entities, 380
- console search view, 381
- Consolidate New Order Releases field, 179
- Consolidate upto volume threshold of field, 65
- Consolidate upto weight threshold of field, 64
- Consolidator field, 73, 74
- Constant field, 419
- Contact Address field, 264
- Conversion Rate From Base field, 362
- Conversion Rate To Base field, 362
- Corporate Address field, 45
- Cost Factor Group to be Used for Landed Cost Calculations field, 55, 60
- Cost Factor Group to be Used for Physical Kit Cost Calculations field, 55
- Cost Factor Group to be Used for Standard Cost Calculations field, 55, 60
- cost factor preferences, 54, 59
- Cost Factor Preferences tab, 54, 60
- count execution process type, 169
- country

- locale, 367
- country codes
 - creating, 348, 489
 - deleting, 350, 491
 - modifying, 349, 490
- Country field, 72, 435
- Create Reservation On Order Creation field, 177
- Create Shipments For Products Included In Work Order field, 180
- currency conversions, 360
 - creating, 361
 - deleting, 363
 - modifying, 363
- currency definitions, 358
 - creating, 359
 - deleting, 360
 - modifying, 360
- Currency field, 367
- custom common code types, 396
 - creating, 396
 - deleting, 398
 - modifying, 398
- custom common code values
 - deleting, 400
 - modifying, 399
- custom common codes
 - adding values to, 398
- custom error codes, 400
 - adding, 401
 - deleting, 402
 - modifying, 401
 - searching, 400
- Customer PO # field, 63, 79

D

- data access policies, 287
- data security component, 873
 - custom validation, 873
 - enterprise validation, 873
- database nodes, 756
 - database receiver properties, 759
 - database sender properties, 757
- database receiver properties, 759
- database sender properties, 757

- Date Format field, 368
- date formats
 - creating, 353
 - deleting, 354
 - modifying, 353
- Date Time Format field, 368
- Date/Hour/Minute Format field, 368
- dates
 - calculations, 101
 - Schedule date calculations, 101
- date/time formats, 356
 - creating, 356
 - deleting, 358
 - modifying, 357
- Day Display Date Format field, 368
- DCS 6.2 database node, 762
 - defining defaults, 763
 - error processing, 763
 - interface tables, 763
 - sender, 763
- DCS 6.2 database nodes
 - DCS 6.2 database receiver properties, 764
 - DCS 6.2 database sender properties, 763
 - receiver, 764
- DCS 6.2 database receiver properties, 764
- DCS 6.2 database sender properties, 763
- Default Declared Value field, 99
- Default Delivery Method Based On Catalog field, 176
- Default Document Type Access field, 284
- Default Effective From field, 156
- Default Effective To field, 156
- Default Enterprise Access field, 283
- Default Label Format, 458
- Default No of Copies field, 54, 59, 84
- defaulting component, 870
 - configuration properties, 871
- Delay shipment by not more than field, 64
- Demand Type for Release field, 177
- Demand Type For Schedule field, 177
- demand types, 177
- Department Abbreviation field, 162
- Department Code field, 63, 79, 263
- departments
 - creating, 161

- deleting, 163
- modifying, 162
- viewing, 161
- Derived Document Type field, 199
- derived transactions
 - creating, 225
- detail view components
 - anchor pages
 - defining, 383
- device Attributes, 448
- device attributes, 450
- Device ID, 448
- device sub types, 448
 - creating, 442
 - defining, 442
 - deleting, 445
 - modifying, 445
- device types, 441, 448
 - creating, 440
 - defining, 439
 - deleting, 442
 - modifying, 441
- devices
 - creating, 446
 - creating new from existing, 451
 - defining, 439, 445
 - deleting, 454
 - device attributes
 - setting up, 451
 - modifying, 454
- dimension unit of measure
 - creating, 332, 334, 336
 - creating conversion rate, 332, 335, 337
 - deleting, 333, 336, 338
 - deleting conversion rate, 333, 336, 338
 - modifying, 332, 334, 337
 - modifying conversion rate, 333, 335, 338
- Dimension UOM field, 367
- Display Error Details field, 273
- Distance Per Day field, 87, 91, 95
- Do not apply SCM field, 78
- Do Not Generate field, 115
- Document Classification field, 175
- document definitions, 47
- Document Description, 457

- Document field, 143, 146, 149
- document format codes
 - creating, 405
 - deleting, 406
 - modifying, 406
- document templates, 180
- Document Type field, 172, 224, 309, 380
- document types, 168, 305
 - configuring, 167
 - modifying description, 173
 - saving as new, 170
- DocumentType attribute, 380
- draft orders, 178
- Driver Date field, 176
- drop statuses, 230
 - adding, 217
 - deleting, 218
- Drop Statuses field, 200
- DUNS Number field, 39, 43
- Dynamic field, 237

E

- Economic shipping parameters maintained field, 64
- Effective Date field, 68
- Effective From field, 362
- Effective Period field, 158
- Effective To field, 362
- EJB nodes, 768
- Electronic Code field, 87, 90, 95
- E-mail nodes, 860
- E-mail Protocol field, 315
- E-mail Server IP Address field, 315
- E-mail Server Listener Port field, 315
- E-mail Server Name field, 315
- E-mail template, 247
- end nodes, 253
- End Time field, 156, 159
- Enterprise Code field, 81
- enterprises, 2, 3, 46, 47
 - defining attributes, 49
 - adding organizations, 51
 - modifying associated organizations, 52
 - removing associated organizations, 53
 - print format preferences, 53

- Entity Key Name field, 388
- environment variable
 - INSTALL_DIR, xlii
 - INSTALL_DIR_OLD, xlii
- error codes
 - customizing, 400
- Escalate Alert After field, 425, 426
- Escalate If Number of Unresolved Alert Exceeds field, 424
- Euro currency, 358
- Euro currency membership, 358
- event handlers
 - defining, 214
- Event ID field, 214
- Event Name field, 214
- events, 200
 - adding to a transaction, 213
 - deleting, 214
 - modifying, 214
- Events field, 200
- Exception Group field, 848
- exception handling, 756, 847
- Exception Reference Name field, 849
- Exception Reference Value field, 849
- Exception References tab, 849
- Exception tab, 846
- exchange rates, 361
- Execute field, 247
- Exemption Certificate field, 152
- Expedite shipment by not more than field, 64
- Expiration Date field, 68
- Export License Number field, 98
- EXPORTTBLPRG purge rule, 302
- Extended Ship Mode field, 87, 91, 95
- extended statuses
 - creating, 230
 - deleting, 232
 - modifying, 231
- extended transactions, 196
- External Application field, 99
- Externally Generated field, 115
- Externally-triggered transactions
 - defining, 201
- externally-triggered transactions, 196

F

- File IO nodes, 769
- File IO receiver node properties, 772
- File IO sender nodes, 770
- File transfer protocol. See FTP
- Fixed Transit Days field, 87, 91, 95
- Format field, 143, 146, 149
- Freight Terms field, 69
- From field, 73
- From Node field, 109
- From Organization field, 417
- From System field, 417
- FTP nodes, 780
- FTP receiver properties, 784
- FTP sender properties, 780
- FTP (File Transfer Protocol), 780

G

- general base document type, 168
- general process type, 169
- Get Capacity Real Time field, 135
- Get External Supply Real Time flag, 125, 130
- Get Inventory Real Time field, 130
- GLN field, 98
- Group Description field, 269
- Group ID field, 269
- Group Name field, 269

H

- Has Freezer field, 87, 91, 95
- Hide Maximum Records Field field, 382
- Hide Navigation Panel field, 383
- Hide Title Bar field, 384
- holds
 - putting on, 316
 - This Transaction Can Be Stopped From Processing Orders That Are On Hold, 200
- Host Code field, 87, 91, 95
- Hour/Minute Time Format field, 368
- HP LaserJet 5P, 442
- HTTP nodes, 788
- Hub, 47, 184

I

- Icon field, 372
- Identified By Parent As field, 98
- Ignore Default API checkbox field, 385
- Ignore Default API field, 382, 383
- Implement As A Java Class field, 225, 309
- Implement As A Service field, 224, 309
- IMPORTTBLPRG purge rule, 302
- In a Synchronous Mode field, 256
- In an Asynchronous Mode field, 256
- inbound shipment process type, 169
- INBOXPRG purge rule, 302
- index
 - catalog search, 532
- Inherit Calendar definition field, 156
- inheritance
 - determining, 14
- Inherited Document Type Access field, 284
- Inherited Enterprise Access field, 283
- initial context factory codes, 322
 - creating, 322
 - deleting, 324
 - modifying, 323
- Initial Context Factory field, 204
- Input Key Name field, 391
- Input Name Space field, 390
- installation rules, 310
- INSTALL_DIR, xlii
- INSTALL_DIR_OLD, xlii
- Integration Service field, 144, 147, 150
- internationalization rules, 347
 - country codes, 347, 489
 - currency conversions, 360
 - currency definitions, 358
 - date formats, 352
 - date/time formats, 356
 - language codes, 350
 - time formats, 354
- inventory consolidation levels, 313
- inventory costing factors, 54, 59
- Inventory Information Not Available To the Selling and Fulfillment Foundation field, 125, 129
- Inventory Is Made Available To the Selling and Fulfillment Foundation field, 125, 129

- Inventory Maintained Within the Selling and Fulfillment Foundation field, 129
- Inventory Monitor Rules tab, 55, 61
- inventory monitoring rules, 55, 60
- Inventory Organization Is field, 130
- inventory organizations, 129, 130
- INVENTORYPRG purge rule, 302
- invoices
 - Pro Forma, 178
- Invoke a Service field, 385
- Invoke an API field, 386
- Invoke following services as part of this action field, 246
- Invoked Services List field, 246
- Is Active field, 214
- Is Reprocessible field, 848
- Is Supervisor field, 264
- ISO-3166 code, 367
- ISO-639 code, 367
- Issuing Authority field, 153
- Item Classification field, 69, 81
- Item ID field, 79
- i18n tag, 380

J

- jasper printer component, 875
 - auditing, 875
 - configuration properties, 876
 - variable file names, 877
- Java Class field, 225, 310
- Java Server Page field, 382, 383, 384, 388
- JavaScript field, 389
- JMS Queue Name field, 203
- JMS Security, 206, 796, 804, 813, 818, 825, 829, 836, 839
 - JMS security properties, 207
 - JBoss, 207, 210
 - WebSphere, 207, 208
- JMS security properties (Queue-based)
 - WebLogic, 207
- jsp list, 378
- JSP tag library
 - i18n, 380

L

Label Format field, 54, 59, 84
Label Format File Name, 461
Label Format Name, 461
language
 locale, 367
language codes
 creating, 351
 deleting, 352
 modifying, 352
Language field, 367
legal entity, 44
Legal Entity field, 44
less-than truckload. See LTL
License Expiration Date field, 98
load base document type, 168
load execution process type, 169
locale
 country, 367
 ISO-3166 code, 367
 ISO-639 code, 367
 language, 367
Locale Code field, 263, 367
Locale Description field, 367
Locale field, 39, 43
locales, 365
 creating, 366
 deleting, 369
 modifying, 368
Loftware Label Manager, 458
Log Audits For Draft Order field, 176
Log File Name field, 304
LTL carrier preferences
 nodes, 113
LTL (Less-than Truck Load), 88

M

Maintain Inventory Cost field, 99
Maintained Externally field, 67
Mandate Serial information for serialized items
 field, 78
Mandate tag information for all tag controlled items
 field, 78

manifesting process type, 169
MANIFESTPRG purge rule, 302
Mapping Type field, 419
Mapping XML File Name, 461
Mark For field, 64, 79
marketplaces, 3, 47
master order fulfillment process type, 169
Max Customer Assignments field, 264
Maximum Length Of Ship Advice Number field, 319
menu configuration, 369
 defining menu items, 371
 modifying application menu details, 370
 saving a menu group as a new menu
 grouping, 369
Menu group, 263
Menu Group field, 263
Menu groups
 saving as a new grouping, 369
menu groups, 369
Menu ID field, 372
menu items
 creating, 371
 deleting, 374
 modifying, 373
menu sequence, 373
Menu Sequence field, 373
monitor rule types, 191
Monitoring rules
 defining, 189
Month Display Date Format field, 368
Move Alert To Different Queue field, 425, 426
move request execution process type, 169
multi-divisional corporations, 2, 47

N

Name field, 68, 141
No. of Threads field, 203
Node field, 74
node properties
 configuring, 257
Node Type field, 98
nodes, 48
 configuring user access, 293
 defining attributes, 96

- defining LTL carrier preferences, 113
- defining node operations, 100
- defining parcel carrier preferences, 104
- defining primary information, 97
- defining PRO Number, 113
- defining roles
 - defining node advanced attributes, 124
- roles
 - nodes, 48
- nomenclature, 6
- nomenclature configuration
 - defining, 416
- nomenclature definitions
 - creating a mapping between, 414
- Nomenclature field, 419
- nomenclature runtime components, 411
 - nomenclature definitions, 414
- Nomenclature runtime nodes, 863
- nomenclature transformation engine, 411
- Non-working day, 160
- Not Maintained field, 66
- Notification Service field, 116
- Notification Threshold field, 116

O

- Only when receiving, 125
- Only when shipping, if the buyer mandates, 125
- order base document type, 168
- Order Creation Tab field, 176
- order fulfillment process type, 168
- order holds
 - deprecated functionality, 316
- order negotiation process type, 168
- order pipeline, 316
- Order # field, 64
- Org Suffix, 43
- Organization Code field, 38, 413
- Organization Defines Its Own Catalog field, 133
- Organization Is An Enterprise field, 39
- organization levels, 14
 - rules, 17
- Organization Name field, 38
- organization rules, 17
 - loading another organization's rules, 21

- overriding, 18
- organizational hierarchies, 164
 - adding organizations to, 164
 - creating, 163
 - removing organizations from, 165
- organizations
 - assigning participant associations, 45
 - assigning roles, 45
 - communication protocols, 140
 - creating, 140
 - deleting, 142
 - modifying, 141
 - communication standards
 - buyer documents, 142
 - creating, 142
 - deleting, 145
 - modifying, 144
 - carrier documents, 148
 - creating, 148
 - deleting, 151
 - modifying, 150
 - communication protocols, 140
 - seller documents, 145
 - creating, 145
 - deleting, 148
 - modifying, 147
- creating, 37
- defining calendars, 154
 - creating an exception, 159
 - defining defaults, 158
- defining payment information, 151
- defining primary information, 42
- defining roles
 - defining Carrier attributes, 84
 - defining carrier attributes
 - defining less-than truckload carrier services, 88
 - defining truckload carrier services, 85
 - setting up parcel carrier services, 92
- defining enterprise attributes, 49
 - adding associated organizations, 51
 - modifying
 - associated organizations, 52
 - removing
 - associated organizations, 53

- defining node attributes, 96
 - defining LTL carrier preferences, 113
 - defining node operations, 100
 - defining parcel carrier preferences, 104
 - defining primary information, 97
 - defining seller attributes, 57, 62
- departments
 - creating, 161
 - deleting, 163
 - modifying, 162
- modifying, 163
- setting up roles
 - setting up enterprise attributes
 - setting up carrier preferences, 54, 59
 - setting up inventory monitoring rules, 55, 60
 - viewing child organizations, 153
 - viewing departments, 161
- Other Currency field, 362
- Other tab, 199
- outbound picking process type, 169
- outbound shipment process type, 168
- Output Name Space field, 382, 383, 384
- over pack build process type, 169
- OverPick For Voice Based Tasks, 127
- Override Entity ID field, 388
- Override Freight Terms field, 74
- Override Modification Rules field, 273
- Override Ship To field, 74

P

- pack process process type, 169
- Package Level Integration field, 88, 91, 96
- packaging parameters, 77
- Packing, 127
- packing slip, 455
- Parameter Name, 451
- Parameter Name field, 206
- Parameter Value, 451
- Parameter Value field, 206
- parcel carrier preferences
 - nodes, 104
- parcels, 92
- Parent Organization field, 43
- Parent Team ID field, 283
- Participant modeling, 37
- participant modeling, 4
 - creating an organization, 37
- participants, 37
- pass-through nodes, 253
- Password field, 263
- Password Policy field, 263
- payment information
 - defining an organization's, 151
- permissions, 270
 - list of users, 271
- PERSONIFOPRG purge rule, 302
- PERSONINFOHISTPRG purge rule, 302
- pickup statuses, 230
 - adding, 215
 - deleting, 217
- pickup Statuses field, 200
- pipeline determination, 184
- pipelines
 - creating, 185
 - defining monitoring rules, 189
 - modifying, 188
- planned order execution process type, 168
- planned order negotiation process type, 169
- Postal Codes Define This Region field, 437
- presentation, 5
- PRICELISTPRG purge rule, 302
- Pricing
 - defining, 137
- Pricing organization, 14, 17
- Primary Enterprise field, 40
- primary enterprises, 49, 140
- Primary URL field, 43
- Print Document field, 54, 59, 84
- print documents, 457
 - creating, 456, 466, 469, 471
 - defining, 456
 - deleting, 459, 469, 470, 476
 - modifying, 458, 468, 470, 472
- print format preferences
 - defining, 53, 58, 83
- Printer ID, 268, 281
- printers, 442

prints

- defining, 455
- label formats
 - creating, 459
 - defining, 459
 - deleting, 462
 - modifying, 461
- Loftware
 - label formats, 459
 - mapping XML file, 459
- Loftware Label Manager, 459
- printer - station association, 265, 279
- printer preferences
 - users
 - setting up, 265
 - user printer preferences
 - defining, 265, 279
- Prior Errors User Exit field, 848
- Priority field, 73
- Pro Forma invoices, 178
- PRO Number End field, 115, 116
- Pro Number Generation Scheme field, 115
- PRO Number Prefix field, 115, 116
- PRO Number Set - 1 field, 115
- PRO Number Set - 2 field, 115
- PRO Number Sets field, 115
- PRO Number Start field, 115, 116
- process modeling, 4, 167
- process modeling tree
 - viewing, 170
- Process Type field, 175
- Process Type Name field, 175
- process type pipelines, 182
- process types, 4, 168, 173
 - count execution, 169
 - defining primary information, 174
 - defining templates, 180
 - general, 169
 - inbound shipment, 169
 - load execution, 169
 - manifesting, 169
 - master order fulfillment, 169
 - move request execution, 169
 - order fulfillment, 168
 - order negotiation, 168
 - outbound picking, 169
 - outbound shipment, 168
 - over pack build, 169
 - pack process, 169
 - planned order execution, 168
 - planned order negotiation, 169
 - purchase order execution, 169
 - purchase order negotiation, 169
 - purchase order receipt, 169
 - return receipt, 169
 - return shipment, 169
 - reverse logistics, 169
 - task execution, 169
 - template order, 169
 - trailer loading, 169
 - transfer order delivery, 169
 - transfer order execution, 169
 - transfer order receipt, 169
 - VAS process, 169
 - WMS inventory, 169
 - WMS layout definition, 169
 - WMS putaway, 169
- Procure To Ship Allowed field, 102
- procurement, 102
 - chained orders, 102
 - purchase orders, 102
 - transfer orders, 102
- Procurement Placed Supply Type field, 178
- procurement purchase orders, 179
- procurement transfer orders, 179
- Product Class field, 79
- Program field, 247
- protocol codes, 403
 - creating, 404
 - deleting, 405
 - modifying, 404
- Protocol field, 141, 143, 146, 149
- Provider URL field, 205
- Publish Data field, 248
- Publish Date field, 68
- purchase order execution process type, 169
- purchase order negotiation process type, 169
- purchase order receipt process type, 169
- Purge Code field, 303
- purge criteria, 301

- modifying the system's, 302
- purges, 301

Q

- QCF Lookup field, 205
- qualified tag types
 - creating, 482
 - deleting, 484
 - modifying, 483
- qualified tag version compatibility
 - configuring, 486
 - deleting, 488
 - modifying, 488
- qualified tags
 - creating, 484
 - deleting, 486
 - modifying, 485
- Queue Description field, 422
- Queue ID field, 422
- Queue management, 421
- queue management, 6, 439, 455, 463, 465, 481
- Queue Name field, 246, 422, 845
- Queue Priority field, 423
- queues, 421
 - based on size, 423
 - creating, 421
 - deleting, 427
 - modifying, 426
 - unassigned alerts, 424
 - unresolved alerts, 425
 - user list, 426

R

- Raise Action field, 424, 425, 426
- Re-Alert After field, 424, 425, 426
- Receipt Processing Time for Forwarding (Hours), 104
- Receipt Processing Time (Hours) field, 104
- receiver, 251
- Receiving, 126
- Receiving Calendar field, 103
- Redirector View field, 384
- region definition, 6

- region definitions, 429
 - region levels, 429
 - region schemas, 433
 - regions, 436
- Region Level Name field, 430
- region levels
 - creating, 430
 - deleting, 432
 - modifying, 431
- Region Name field, 436
- Region Schema For Routing field, 66
- Region Schema Name field, 435
- region schemas
 - creating, 434
 - deleting, 438
 - modifying, 438
- regions
 - creating, 436, 437
 - deleting, 437
- related entity, 381
- Related Entity field, 381
- repositories, 182, 183
- REPROCESSPRG purge rule, 302
- Requested Delivery Date field, 176
- Requested Ship Date field, 176
- Requires Backward Compatibility field, 214, 225, 310
- Resource ID, 373
- Resource ID field, 373, 376
- Resource Identifier, 43
- Resource Planning Enabled field, 99
- Resource Sequence field, 378
- resource sequencing, 378
- Resource Type field, 377
- resources, 374
 - console actions, 389
 - console APIs, 385
 - console detail views
 - console detail views, 382
 - console entities, 378
 - viewing jsp list, 378
 - viewing user list, 378
 - console icons, 392
 - console inner panels, 388
 - console links, 391

- console list views, 384
- console related entities, 380
- console search view, 381
- Restrict Access To A Specific List Of Enterprises field, 284
- Restrict List To A Specific List Of Document Types field, 285
- Retention Days field, 303
- return receipt process type, 169
- return shipment process type, 169
- reverse logistics process type, 169
- RF Scanners, 442
- roles
 - assigning to a yard, 139
 - assigning to an organization, 48
 - buyers, 46
 - multiple organizations, 47
 - organization, 45
 - sellers, 48
- Roles & Participation tab, 49
- Rollback Segment field, 303
- router nodes, 865
- routing guide lines, 70
- Routing Guide # field, 68
- routing guides, 65
 - fields, 66
 - VICS, 66
- Run Quantity field, 82
- Runtime ID field, 845
- Runtime tab, 845

S

- Save Directory, 458
- SCM not required field, 78
- SCM required with content field, 78
- SCM required without content field, 78
- security, 4, 261
 - data access policies, 287
- security management, 261
- Select Algorithm field, 116
- Selection Key Name field, 390
- seller, 145
- seller documents, 145, 147, 148, 409
- Seller Documents tab, 145

- sellers, 3, 48, 148
 - configuring data access, 293
 - defining attributes, 57
 - print format preferences, 58
- Send E-Mail field, 247
- Send Fax field, 248
- Send to Alert Console field, 246
- sender, 251
- Server Name field, 846
- Server tab, 846
- service definitions, 250, 305
 - creating, 254
- Service Description field, 86, 90, 94
- Service Group Name field, 246, 255
- Service Name field, 225, 246, 255, 310, 385
- service nodes, 250
- Set up data for System Generated scheme field, 115
- Shift Name field, 156, 158
- Shift Valid For field, 159
- shifts, 103, 157
- Ship By Air field, 87, 91, 95
- Shipment Level Integration field, 88, 92, 96
- Shipping, 127
- Shipping Calendar field, 104
- shipping labels, 455
- Skip Automatic Execution field, 387
- sourcing region selection
 - defining, 493
- Spawning process type field, 199
- Spawns another process? field, 199
- start nodes, 253
- Start Time field, 156, 159
- State field, 72
- Static field, 237
- STATISTICSPRG purge rule, 302
- status change listeners
 - creating, 226
- status monitoring rule, 232
- status monitoring rule definitions
 - defining, 232
- statuses, 230, 316
- Sterling Order Management, 99
- Sterling WMS, 99
- Stored Procedure field, 248

- Store# field, 73, 74
- Subscribed field, 264, 292
- Substitution Is Allowed field, 102
- Supersedes Routing Guide # field, 68
- Supports Integration for Return Shipping Label field, 88
- Suspend API field, 847
- Suspend Wait Time field, 848
- synchronous MQSeries nodes, 794, 838, 839
- synchronous transport nodes, 251
- synchronous WebLogic nodes, 794
- System Generated field, 115
- System ID field, 248
- System Name field, 413

T

- task execution process type, 169
- Tax Exempt field, 152
- Tax Jurisdiction field, 153
- Tax Payer ID field, 152
- Team field, 264
- Team ID field, 283
- teams, 264, 282
 - creating, 282
 - deleting, 287
 - modifying, 286
- Template Document Type field, 176
- Template field, 247, 248
- template order process type, 169
- templates, 180
- text translator components
 - schema files, 884
 - XSD files, 883
 - delimited receiver, 896
 - delimited sender, 902
 - positional receiver, 889
 - positional sender, 899
- text translator file formats
 - delimited, 882
 - positional, 881
 - XML, 882
- text translator nodes, 866
- Theme field, 263, 395
- themes, 394

- creating, 394
- deleting, 396
- modifying, 395
- Third Party Logistics Node field, 99
- third-party logistics, 47
 - nodes, 99
- third-party logistics models, 3
- This Organization Is A Capacity Organization field, 134
- This Organization Is An Inventory Organization field, 129, 130
- This Region Level Can Be Defined By A Set Of Postal Codes field, 431
- This Service Provides Real-Time Response field, 255
- This Transaction Can Be Stopped From Processing Orders That Are On Hold field, 200
- This Transaction Is Derived From Abstract field, 199
- This transaction is task based field, 199
- Time, 479
- Time Format field, 368
- time formats
 - creating, 354
 - deleting, 356
 - modifying, 355
- time unit of measure
 - creating, 344
 - creating conversion rate, 345
 - deleting, 346
 - deleting conversion rate, 345
 - modifying, 344
 - modifying conversion rate, 345
- Time Zone field, 367
- time-triggered transactions, 196
 - defining, 202
- To field, 73
- To Node field, 109
- To Organization field, 417
- To System field, 417
- trailer loading process type, 169
- transaction completion, 218
 - events, 221
 - status based transactions, 219
 - supported transactions

- custom, 218
 - extended, 218
- Transaction ID field, 198
- Transaction Name field, 198
- transactions, 167, 195
 - adding events, 213
 - creating, 196
 - creating a derived transaction, 225
 - creating a status change listener, 226
 - defining as externally-triggered, 201
 - defining as user-triggered, 212
 - defining event handlers, 215
 - deleting, 229
 - deleting an event, 214
 - drop statuses
 - adding, 217
 - deleting, 218
 - externally-triggered, 196
 - managing user exits, 222
 - modifying, 229
 - modifying an event, 214
 - pickup statuses
 - adding, 216
 - deleting, 217
 - time-triggered, 196
 - defining, 202
 - user-triggered, 196
- transfer order delivery process type, 169
- transfer order execution process type, 169
- transfer order receipt process type, 169
- transfer orders, 108
- transfer schedules, 108
- transport nodes, 250, 755
- truckloads, 85
- tThird-party logistics models, 2

U

- UCC-128, 455
- unit, 345
- Unit Of Measure field, 79
- Units of Measure. See UOM
- UOMs (Units of Measure)
 - defining, 331, 334, 336, 339, 341, 344
 - dimension, 336

- modifying, 332
 - Quantity, 331
 - service quantity, 334
 - volume, 339
- UPS Standard carrier labels, 455
- URL field, 248
- Use as a Template Order field, 175
- Use Default Values field, 419
- Use End Of Shift Time field, 103
- Use Numeric Ship Advice Number flag, 318
- Use Template Order For Defaulting field, 176
- Use units of measure from buyer catalog field, 78
- user, 261
 - configuring data access, 288
 - creating and modifying, 262
 - deleting, 268
- User Exit Implementation Notes field, 225, 310
- user exit implementations
 - defining, 306
- user exits, 222, 304
- user groups, 268
 - administering resource permissions, 270
 - creating and modifying, 269
 - deleting, 282
 - viewing subscribed users, 279
- User ID field, 246, 263
- user interface extensibility, 5
- user list, 378
- User Name field, 263
- USERACTAUDITPRG purge rule, 302
- USERACTIVITYPRG purge rule, 302
- user-triggered transactions, 196
 - defining, 212

V

- Validate Item During Order Creation/ Modification field, 175
- Value field, 141
- VAS process type, 169
- Version field, 225, 310
- Version Label Id, 479
- Version Label Id Description, 479
- VICS, 66
- VICS Bill Of Lading (BOL), 455

View Group ID field, 382, 384, 385, 390
View ID field, 390
View Values field, 414
volume unit of measure
 creating, 339
 creating conversion rate, 340
 deleting, 341
 deleting conversion rate, 341
 modifying, 339
 modifying conversion rate, 340
Volume UOM field, 367

W

wave release prints document set, 456
WebServices nodes, 791
weight, 341
weight unit of measure
 creating, 341
 creating conversion rate, 342
 deleting, 343
 deleting conversion rate, 343
 modifying, 342
 modifying conversion rate, 343
Weight UOM field, 368
When Agent/Integration Server Terminates
 Unexpectedly field, 325
When Application Server Goes Down field, 325
When Monitoring Threshold Reached or Exceeded
 field, 325
When performing any node operation, 125
WMS inventory process type, 169
WMS layout definition process type, 169
WMS putaway process type, 169
WMS 6.2 field, 99
Work Day Hours field, 99
Work Order Creation Is Allowed field, 102
workflows, 167
working days, 160
Works based on data from field, 199
Write to Log File field, 304

X

XML binding, 390, 392

XML Name field, 417, 418
XML templates, 180
XSL translator nodes, 868

Z

Zebra 170, 442
Zip Code field, 72