

Sterling Commerce

An IBM Company

Catalog Management: Configuration Guide

Release 8.5

Last updated in HF22

August 2010



Copyright Notice

Copyright © 1999 - 2010

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 - 2010

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	xxix
Structure	xxix
Selling and Fulfillment Foundation Documentation	xxx
Conventions	xxxii

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 Catalog Management Configuration	4
1.2.1 Catalog Configurations.....	4
1.2.2 Products	5
1.2.3 Delivery Services	5
1.2.4 Provided Services.....	5
1.2.5 Value Added Services	5

2 Navigating the Applications Manager

2.1 Starting the Applications Manager	7
2.2 The Applications Manager Layout	8
2.2.1 Application Rules Side Panel	10
2.2.1.1 Accessing Configuration Screens	11
2.2.1.2 Determining Inheritance.....	12

2.2.1.3	Loading Another Organization's Rules	19
2.2.2	Work Area	21
2.2.2.1	Search Window	21
2.2.2.2	List Window	22
2.2.2.3	Details Window	23
2.2.2.4	Drag and Drop Window	24
2.3	Actions Available in the Applications Manager	26
2.3.1	Using the Applications Manager's Lookup Functionality	26
2.3.2	Viewing the Document Types Associated with an Application	27
2.3.2.1	Adding a Document Type to an Application	28
2.3.3	Viewing the User Logged into the Applications Manager	29
2.3.4	Using Lists and List Filtering	29
2.3.5	Date and Time Entry	32
2.3.6	Using On-Line Help	32
2.3.7	Troubleshooting Errors	32
2.3.8	Using Special Characters	33

3 Configuring Catalog Components

3.1	Defining Item Statuses	35
3.1.1	Creating an Item Status	36
3.1.2	Modifying an Item Status	37
3.1.3	Deleting an Item Status	37
3.2	Defining Additional Attributes	37
3.2.1	Creating an Additional Attribute	38
3.2.2	Modifying an Additional Attribute	38
3.2.3	Deleting an Additional Attribute	39
3.3	Defining Item Instruction Codes	39
3.3.1	Creating an Item Instruction Code	39
3.3.2	Modifying an Item Instruction Code	41
3.3.3	Deleting an Item Instruction Code	41
3.4	Defining Item Instruction Types	41
3.4.1	Creating an Item Instruction Type	42
3.4.2	Modifying an Item Instruction Type	43
3.4.3	Deleting an Item Instruction Type	43
3.5	Defining Service Skills	43

3.5.1	Creating a Service Skill	44
3.5.2	Modifying a Service Skill.....	45
3.5.3	Deleting a Service Skill	46
3.6	Defining Service Complexity Levels.....	46
3.6.1	Creating a Service Complexity Level	46
3.6.2	Modifying a Service Complexity Level.....	47
3.6.3	Deleting a Service Complexity Level.....	48
3.7	Defining Item Alias Types.....	48
3.7.1	Creating an Item Alias Type.....	48
3.7.2	Modifying an Item Alias Type	49
3.7.3	Deleting an Item Alias Type.....	50
3.8	Defining Item Association Types.....	50
3.8.1	Creating an Item Association Type.....	50
3.8.2	Modifying an Item Association Type	51
3.8.3	Deleting an Item Association Type	52
3.9	Defining Item Exclusion Codes	52
3.9.1	Creating an Item Exclusion Code	53
3.9.2	Modifying an Item Exclusion Code	54
3.9.3	Deleting an Item Exclusion Code	54
3.10	Defining Catalog Rules	54
3.11	Defining Assets	55
3.11.1	Creating an Asset Type	56
3.11.2	Modifying an Asset Type.....	57
3.11.3	Deleting an Asset Type	57
3.11.4	Creating an Asset Content Location	58
3.11.5	Modifying an Asset Content Location.....	58
3.11.6	Deleting an Asset Content Location.....	59

4 Configuring Products

4.1	Defining Product Items	61
4.1.1	Creating a Product Item	62
4.1.2	Modifying a Product Item	63
4.1.2.1	Defining a Product Item's Primary Information	64
4.1.2.2	Defining a Product Item's Units of Measure	68
4.1.2.2.1	Adding an Alternate Unit of Measure to a Product Item.....	71

4.1.2.2.2	Modifying a Product Item's Alternate Unit of Measure	73
4.1.2.2.3	Removing a Product Item's Alternate Unit of Measure.....	73
4.1.2.3	Defining a Product Item's Sourcing and Transportation Info	73
4.1.2.4	Defining a Product Item's Inventory Information.....	76
4.1.2.5	Viewing Inheritable Attributes	81
4.1.2.6	Defining a Product Item's Classifications	82
4.1.2.7	Defining a Product Item's Additional Attributes	86
4.1.2.7.1	Modifying a Product Item Instruction.....	89
4.1.2.7.2	Deleting a Product Item Instruction.....	89
4.1.2.8	Defining Product Item Associations	89
4.1.2.8.1	Association by Item	91
4.1.2.8.2	Association by Query	92
4.1.2.9	Defining a Product Item's Node Item Attribute Overrides	94
4.1.2.10	Defining a Product Item's Associated Service Items.....	94
4.1.2.10.1	Associating a Service Item with a Product Item	95
4.1.2.10.2	Modifying Service Items Associated with a Product Item	97
4.1.2.10.3	Deleting a Service Item Associated with a Product Item.....	97
4.1.2.10.4	Adding a Service Skill to a Service Association.....	98
4.1.2.10.5	Removing a Service Skill from a Service Association.....	98
4.1.2.11	Defining a Product Item's Instructions	98
4.1.2.11.1	Adding a Product Item Instruction	98
4.1.2.11.2	Modifying a Product Item Instruction.....	100
4.1.2.11.3	Deleting a Product Item Instruction.....	100
4.1.2.12	Defining a Product Item's Children Items	100
4.1.2.12.1	Adding a Child Item	101
4.1.2.12.2	Removing a Child Item.....	101
4.1.2.13	Defining a Product Item's Components	101
4.1.2.13.1	Adding a Component to an Item	102
4.1.2.13.2	Modifying an Item's Component	104
4.1.2.13.3	Deleting an Item's Component	105
4.1.2.14	Defining a Product Item's Container Attributes	105
4.1.2.14.1	Adding a SKU Capacity Definition.....	108
4.1.2.14.2	Modifying a SKU Capacity Definition	109
4.1.2.14.3	Deleting a SKU Capacity Definition.....	109
4.1.2.15	Defining a Product Item's Extended Attributes	110

4.1.3	Deleting an Item	111
4.1.4	Recomputing an Item's Computed Unit Cost	111
4.1.5	Using the Condition Builder.....	111
4.1.5.1	Using the Condition Builder.....	112
4.1.5.2	Condition Builder Attributes	116
4.2	Defining Item Attribute Overrides at the Node Level.....	120
4.2.1	Creating Item Attribute Overrides at the Node Level	120
4.2.2	Modifying Item Attribute Overrides at the Node Level.....	122
4.2.3	Deleting Item Attribute Overrides at the Node Level.....	122
4.3	Defining Master Units of Measure	122
4.3.1	Defining Quantity Units of Measure	123
4.3.1.1	Creating a Quantity Unit of Measure.....	123
4.3.1.2	Modifying a Quantity Unit of Measure	125
4.3.1.3	Deleting a Quantity Unit of Measure	126
4.3.2	Defining Pricing Units of Measure.....	126
4.3.2.1	Creating a Pricing Unit of Measure	126
4.3.2.2	Modifying a Pricing Unit of Measure.....	127
4.3.2.3	Deleting a Pricing Unit of Measure	127
4.4	Defining Master Catalogs.....	128
4.4.1	Creating a Master Catalog	128
4.4.2	Modifying a Master Catalog.....	129
4.4.3	Deleting a Master Catalog.....	130
4.5	Defining Categories	130
4.5.1	Defining Additional Category Attributes	130
4.5.1.1	Defining Category Statuses.....	131
4.5.1.1.1	Creating a Category Status.....	131
4.5.1.1.2	Modifying a Category Status	132
4.5.1.1.3	Deleting a Category Status	132
4.5.1.2	Defining Additional Attributes.....	132
4.5.1.2.1	Creating an Additional Attribute.....	133
4.5.1.2.2	Modifying an Additional Attribute	134
4.5.1.2.3	Deleting an Additional Attribute.....	134
4.6	Defining Classifications	135
4.6.1	Defining Classification Definitions	136
4.6.1.1	Creating a Classification Definition	137

4.6.1.2	Modifying a Classification Definition	138
4.6.1.3	Deleting a Classification Definition	138
4.6.2	Defining Classification Purposes.....	138
4.6.2.1	Creating a Classification Purpose	141
4.6.2.2	Modifying a Classification Purpose.....	142
4.6.2.3	Deleting Classification Purposes.....	142
4.6.3	Defining a Classification Hierarchy	143
4.6.3.1	Creating a Classification Value.....	143
4.6.3.2	Modifying and Adding Items to a Classification Value	145
4.6.3.3	Modifying and Adding Items to a Classification Value	146
4.6.3.4	Defining Common Item Attributes for a Classification Value.....	146
4.6.3.5	Deleting a Classification Value.....	151
4.6.3.6	Associating Provided Services and Delivery Services to a Classification Value.....	151
4.6.3.6.1	Creating a Provided Service Association.....	153
4.6.3.6.2	Modifying a Provided Service Association	154
4.6.3.6.3	Deleting a Provided Service Association	154
4.6.3.6.4	Creating a Delivery Service Association	154
4.6.3.6.5	Modifying a Delivery Service Association.....	155
4.6.3.6.6	Deleting a Delivery Service Association.....	155
4.7	Defining Item Attributes.....	156
4.7.1	Defining Alias Types	156
4.7.1.1	Creating an Alias Type	156
4.7.1.2	Modifying an Alias Type.....	157
4.7.1.3	Deleting an Alias Type.....	158
4.7.2	Item Cost Posting Classification.....	158
4.7.2.1	Creating an Item's Cost Posting Classification.....	158
4.7.2.2	Modifying an Item's Cost Posting Classification	159
4.7.2.3	Deleting an Item's Cost Posting Classification.....	160

5 Configuring Delivery Services

5.1	Defining Delivery Service Options	162
5.1.1	Creating a Delivery Service Option.....	162
5.1.2	Modifying a Delivery Service Option	164
5.1.3	Deleting a Delivery Service Option.....	164

5.2	Defining Delivery Service Master Units of Measure	164
5.2.1	Creating a Master Unit of Measure	165
5.2.2	Modifying a Master Unit of Measure	165
5.2.3	Deleting a Master Unit of Measure	166
5.3	Defining Service Types.....	166
5.3.1	Defining Delivery Service Types	166
5.3.1.1	Creating a Delivery Service Type	166
5.3.1.2	Modifying a Delivery Service Type	168
5.3.1.3	Deleting a Delivery Service Type	168
5.3.2	Defining Additional Capacity	168
5.3.2.1	Creating Additional Capacity	168
5.3.2.2	Modifying Additional Capacity.....	170
5.3.2.3	Deleting Additional Capacity	170

6 Configuring Provided Services

6.1	Defining Provided Service Options	172
6.1.1	Creating a Provided Service Option.....	172
6.1.2	Modifying a Provided Service Option	174
6.1.3	Deleting a Provided Service Option	174
6.2	Defining Provided Service Master Units of Measure	174
6.2.1	Creating a Master Unit of Measure	175
6.2.2	Modifying a Master Unit of Measure	175
6.2.3	Deleting a Master Unit of Measure	176
6.3	Defining Provided Service Types.....	176
6.3.1	Creating a Provided Service Type	176
6.3.2	Modifying a Provided Service Type.....	178
6.3.3	Deleting a Provided Service Type.....	178

7 Configuring Value Added Services

7.1	Defining a Value Added Service.....	180
7.1.1	Creating a Value Added Service.....	181
7.1.1.1	Defining a Value Added Service's Primary Information.....	181
7.1.1.2	Defining a Value Added Service's Item Instructions.....	184
7.1.1.2.1	Creating Item Instructions.....	185
7.1.1.2.2	Modifying an Item Instruction	187

7.1.1.2.3	Deleting an Item Instruction	187
7.1.1.3	Defining a Value Added Service's Service Activities	187
7.1.1.3.1	Creating a Service Activity	188
7.1.1.3.2	Modifying a Service Activity	190
7.1.1.3.3	Deleting a Service Activity	190
7.1.2	Modifying a Value Added Service	191
7.1.3	Deleting a Value Added Service	191
7.2	Defining an Activity	191
7.2.1	Creating an Activity	191
7.2.2	Modifying an Activity.....	192
7.2.3	Deleting an Activity	192
A.1	Running Time-Triggered Transactions	195
A.1.1	Steps to Complete Before Scheduling Time-Triggered Transactions ...	195
A.2	Configuring Communication Between an Agent and a JMS Server.....	196
A.2.1	Prerequisites.....	196
A.2.2	Create an Initial Context Factory Code	197
A.2.3	Define the Transaction Information.....	198
A.3	Business Process Time-Triggered Transactions	199
A.3.1	Asynchronous Request Processor.....	200
A.3.2	Case Insensitive Data Loader	202
A.3.3	Change Load Status.....	204
A.3.4	Change Shipment Status	206
A.3.5	Close Delivery Plan	208
A.3.6	Close Load	210
A.3.7	Close Manifest	212
A.3.8	Close Order	215
A.3.9	Close Receipts	217
A.3.10	Close Shipment	219
A.3.11	Collect Shipment Statistics	221
A.3.12	Consolidate Additional Inventory	224
A.3.13	Consolidate To Shipment.....	226
A.3.14	Create Catalog Index	230
A.3.15	Create Chained Order	234
A.3.16	Create Derived Order	236
A.3.17	Create Order Invoice	238

A.3.18	Create Shipment Invoice	240
A.3.19	ESP Evaluator	242
A.3.20	Item Based Allocation	245
A.3.21	Mark Load as Trailer Loaded	251
A.3.22	Match Inventory.....	253
A.3.23	Payment Collection.....	255
A.3.24	Payment Execution.....	258
A.3.25	Post Inventory Match.....	261
A.3.26	Process Order Hold Type	263
A.3.27	Process Work Order Hold Type	266
A.3.28	Publish Negotiation Results.....	268
A.3.29	Release.....	270
A.3.30	Route Shipment	273
A.3.31	Schedule.....	276
A.3.32	Send Invoice	280
A.3.33	Send Item Changes	283
A.3.34	Send Customer Changes	285
A.3.35	Send Order	287
A.3.36	Send Release.....	289
A.3.37	Start Order Negotiation.....	291
A.3.38	Synchronize Colony Map	292
A.3.39	Update Best Match Region	294
A.3.40	PopulateOwnershipTransferSummary	297
A.4	Time-Triggered Purge Transactions.....	298
A.4.1	Purge Strategy.....	299
A.4.2	Configuring Purge Transaction Log Files.....	299
A.4.3	Available Purges.....	300
A.4.3.1	Alert Purge.....	301
A.4.3.2	Capacity Purge.....	304
A.4.3.3	Draft Order History Purge.....	307
A.4.3.4	Draft Order Purge.....	310
A.4.3.5	Delivery Plan Purge	314
A.4.3.6	Export Table Purge	317
A.4.3.7	Import Table Purge.....	319
A.4.3.8	Inventory Audit Purge.....	322

A.4.3.9	Inventory Purge	325
A.4.3.10	Inventory Supply Temp Purge	328
A.4.3.11	Item Audit Purge	330
A.4.3.12	Load History Purge	332
A.4.3.13	Load Purge	335
A.4.3.14	Negotiation History Purge	338
A.4.3.15	Negotiation Purge	340
A.4.3.16	Order History Purge	343
A.4.3.17	Order Purge	346
A.4.3.18	Order Release Status Purge	353
A.4.3.19	Order Status Audit Purge	356
A.4.3.20	Organization Audit Purge	358
A.4.3.21	Person Info Purge	360
A.4.3.22	Person Info History Purge	363
A.4.3.23	Picklist Purge	366
A.4.3.24	Price List Purge	368
A.4.3.25	Purge Catalog Mass Audits	370
A.4.3.26	Receipt History Purge	372
A.4.3.27	Receipt Purge	375
A.4.3.28	Reprocess Error Purge	378
A.4.3.29	Reservation Purge	380
A.4.3.30	Shipment History Purge	382
A.4.3.31	Shipment Purge	384
A.4.3.32	Shipment Statistics Purge	388
A.4.3.33	Statistics Purge	390
A.4.3.34	User Activity Purge	393
A.4.3.35	User Activity Audit Purge	395
A.4.3.36	Work Order History Purge	397
A.4.3.37	Work Order Purge	400
A.4.3.38	YFS Audit Purge	403
A.4.3.39	YFSInventoryOwnershipAudit Purge	406
A.4.3.40	Password Reset Request Purge	407
A.4.3.41	User Login Failure Purge	409
A.5	Task Queue Syncher Time-Triggered Transactions	411
A.5.1	Load Execution Task Queue Syncher	412

A.5.2	Order Delivery Task Queue Syncher	414
A.5.3	Order Fulfillment Task Queue Syncher.....	415
A.5.4	Order Negotiation Task Queue Syncher	416
A.6	Monitors	418
A.6.1	Availability Monitor	418
A.6.2	Exception Monitor	420
A.6.3	Inventory Monitor	423
A.6.4	Negotiation Monitor	425
A.6.5	Enhanced Order Monitor.....	427
A.6.6	Enhanced Return Monitor	431
A.6.7	Real-time Availability Monitor.....	435
A.6.8	Shipment Monitor.....	441
A.6.9	Work Order Monitor	444

Index

Preface

This manual describes how to use the Applications Manager for Catalog Management configuration.

Intended Audience

This manual is intended for use by system administrators and managers who need to configure the Selling and Fulfillment Foundation Catalog Management rules, items, categories, and so on, in order to define a product catalog and the services associated with the products, that can be used in other applications within Selling and Fulfillment Foundation.

Structure

This manual contains the following sections:

Chapter 1, "Introduction"

This chapter briefly describes the contents of this guide.

Chapter 2, "Navigating the Configurator"

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 Catalog Components"

This chapter explains how you can configure the common codes used throughout the Catalog Management application and Business Center application.

Chapter 4, "Configuring Products"

This chapter explains how you can configure products in Selling and Fulfillment Foundation. A product item is a physical unit that can be ordered, shipped, and returned.

Chapter 5, "Configuring Delivery Services"

This chapter explains how you can configure delivery services. A delivery service is a service item that is used for last mile delivery associations with product items.

Chapter 6, "Configuring Provided Services"

This chapter explains how you can configure provided services. A provided service is any additional action that can be provided in an order that does not represent a physical product item.

Chapter 7, "Configuring Value Added Services"

This chapter explains how you can configure value added services in Selling and Fulfillment Foundation.

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.

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.
<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>.

Convention	Meaning
<YFS_HOME_OLD>	<p>This is the <YANTRA_HOME>/Runtime directory for Releases 7.7, 7.9, or 7.11.</p> <p>Note: This is only applicable for users upgrading from Releases 7.7, 7.9, or 7.11.</p>
<ANALYTICS_HOME>	<p>User-supplied location of the Sterling Analytics installation directory.</p> <p>Note: This convention is used only in the <i>Selling and Fulfillment Foundation: Business Intelligence Guide</i>.</p>
<COGNOS_HOME>	<p>User-supplied location of the IBM Cognos 8 Business Intelligence installation directory.</p> <p>Note: This convention is used only in the <i>Selling and Fulfillment Foundation: Business Intelligence Guide</i>.</p>
<MQ_JAVA_INSTALL_PATH>	<p>User-supplied location of the IBM WebSphere® MQ Java components installation directory.</p> <p>Note: This convention is used only in the <i>Selling and Fulfillment Foundation: System Manangement and Administration Guide</i>.</p>
<DB>	<p>Refers to Oracle®, IBM DB2®, or Microsoft SQL Server® depending on the database server.</p>
<DB_TYPE>	<p>Depending on the database used, considers the value oracle, db2, or sqlserver.</p>

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 Catalog Management business application in 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 see the *Selling and Fulfillment Foundation: Customization Guide* for information about extending Selling and Fulfillment Foundation. System Integrators should refer to the *Selling and Fulfillment Foundation: Integration Guide* for information about extending or 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 *Item Concepts Guide*.

The Applications Manager is a collection of all the rules and setup configurations necessary to implement Selling and Fulfillment Foundation organized so 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 Catalog Management Configuration

The Catalog Management application enables your organization to control the information and attributes associated with items and catalogs.

NOTE: If you are running a Distributed Order Management deployment, use the Sterling Business Center to configure master data, such as items, categories, and classifications. See the *Item Administration Guide* for more information. Additionally, use the Catalog Management application to perform one-time maintenance tasks related to Catalog Management, such as defining configuration data.

If you are running a Sterling Warehouse Management System deployment, use the Catalog Management application to configure master data in addition to configuration data. Refer to this *Catalog Management: Configuration Guide* for instructions.

You can use the Catalog Management configuration grouping to configure the following aspects of Selling and Fulfillment Foundation for your business application modules:

- [Catalog Configurations](#)
- [Products](#)
- [Delivery Services](#)
- [Provided Services](#)
- [Value Added Services](#)

1.2.1 Catalog Configurations

You can define the common codes used throughout the Catalog Management application.

For more information about Catalog Configurations, see [Chapter 3, "Configuring Catalog Components"](#).

1.2.2 Products

A product item is a physical unit that can be ordered, shipped, and returned. A product item is uniquely defined by its item ID and unit of measure. You can define product item attributes, as well as the catalogs they belong to.

For more information about Products, see [Chapter 4, "Configuring Products"](#).

1.2.3 Delivery Services

A delivery service is a service item that is used for last mile delivery associations. Last mile delivery differs from order shipping in that specific product item deliveries are made using a carrier with defined capacity constraints. You can define delivery service item attributes, options, and units of measure.

For more information about Delivery Services, see [Chapter 5, "Configuring Delivery Services"](#).

1.2.4 Provided Services

A provided service is any additional action that can be provided in an order that does not represent a physical product item. An installation service for a washing machine product item is an example of a provided service. You can define provided service item attributes, options, and units of measure.

1.2.5 Value Added Services

Value added services are services that customize a product for a buyer, or allow for internal inventory management. For example, a buyer may require that a special label is applied to certain items. This can be done before the item is sent to the customer.

You can define Kitting services, DeKitting services, Inventory Change services, and Compliance services.

For more information about Value Added Services, see [Chapter 7, "Configuring Value Added Services"](#).

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 Configurator. 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 Configurator's menu bar.

Figure 2–1 Applications Menu



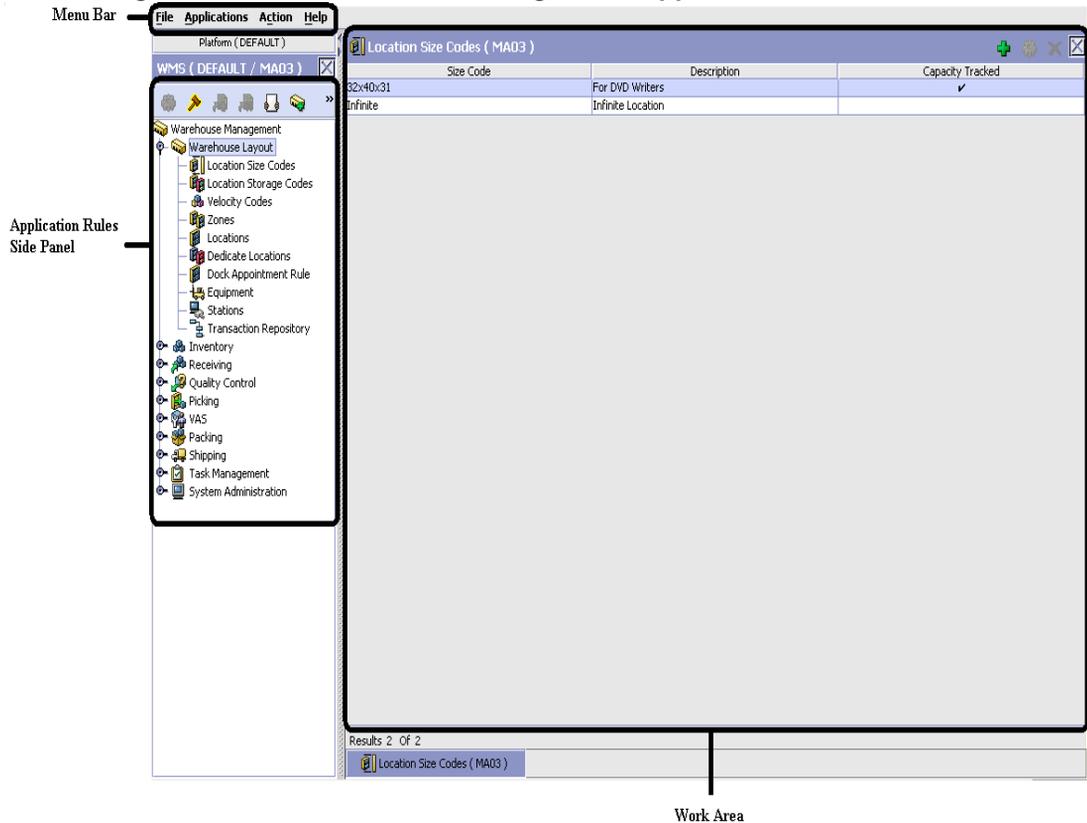
Each application focuses on a particular aspect of Selling and Fulfillment Foundation and contains all of the rules, common codes, and settings necessary for Selling and Fulfillment Foundation to work in a real-world business setting.

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 Configurator displays a side panel containing all of the available configuration rules for the selected application and a work area in which these rules can be configured.

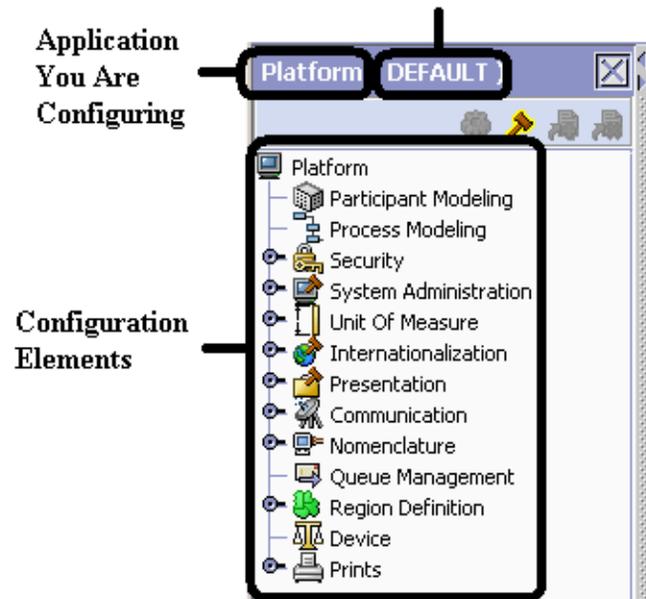
Figure 2–2 The Standard Configurator Application Interface



2.2.1 Application Rules Side Panel

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

Figure 2–3 Example of Application Rules Side Panel 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

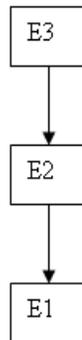
Note: The Configurator does not load configuration data and permissions based on Data Access Policies that are described in the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

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.

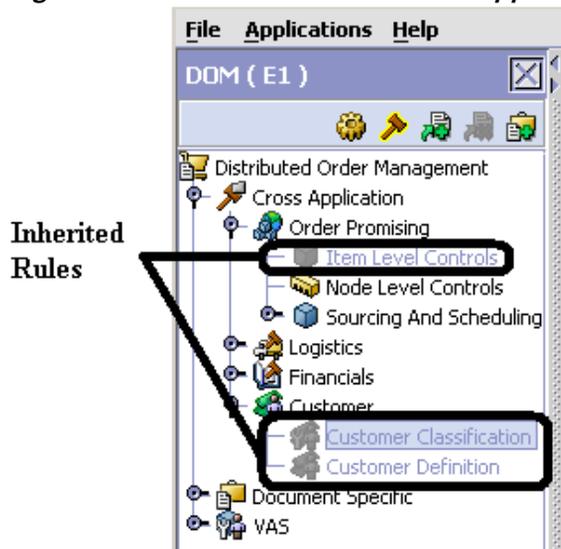
Table 2–1 Organization Level Rules

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.

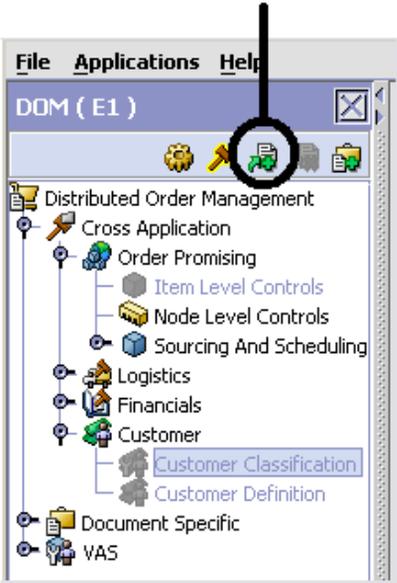
The application rules side panel displays rules that have been inherited as grayed out.

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



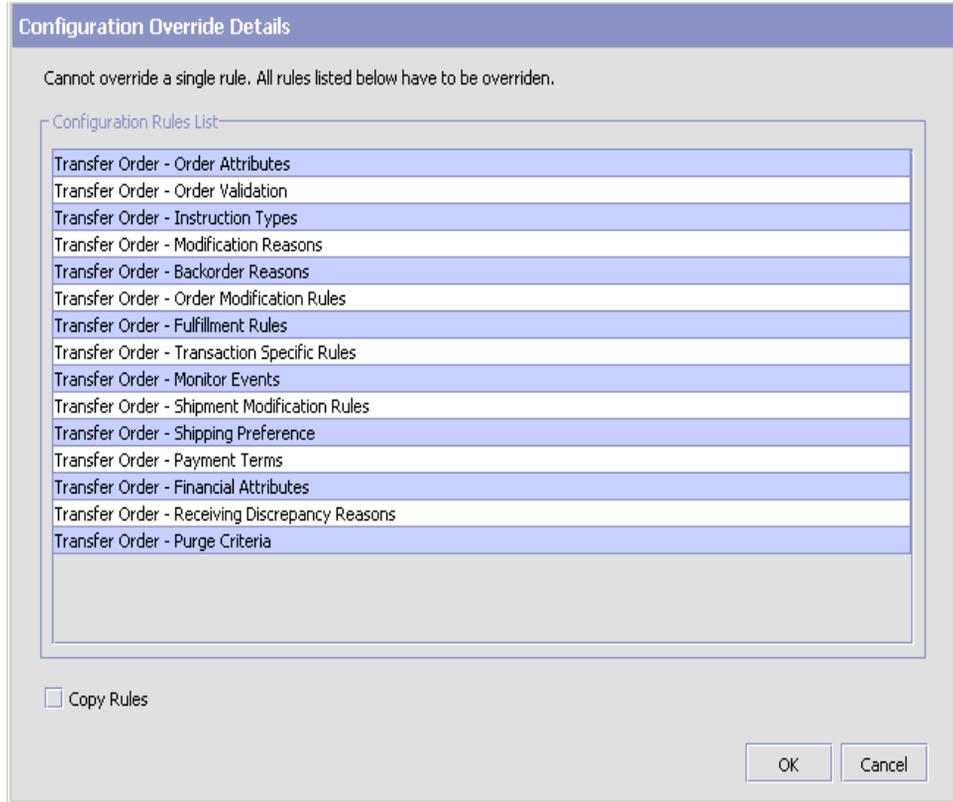
As stated in the table above, depending on the organization you are logged in as, you may be able to override some inherited rules. If a rule can be overridden, the Override Configuration icon becomes available in the application rule side panel when you highlight the rule.

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



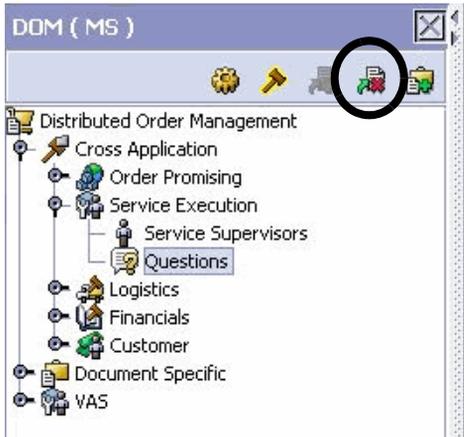
When you choose to override a rule you also override any other rules in the configuration group the rule you are overriding is associated with. When you choose the Override Configuration icon the Configuration Override Details pop-up window displays. This window provides the list of rules that are overridden.

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.

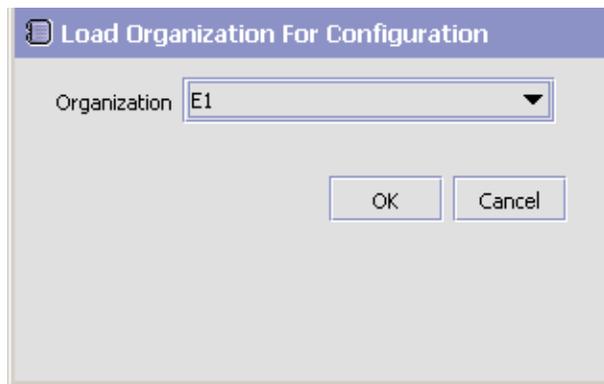
2.2.1.3 Loading Another Organization’s Rules

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

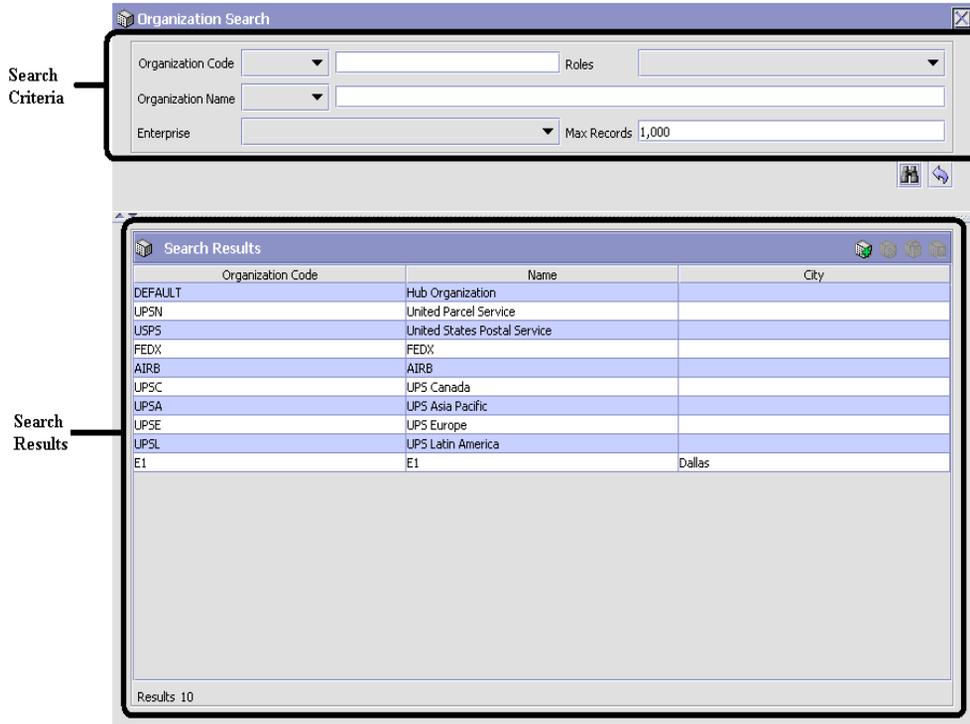
The work area is the main area in which different configuration screens appear. The following are the main types of screens that you can be seen in the work area:

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

Figure 2-9 List Window Example

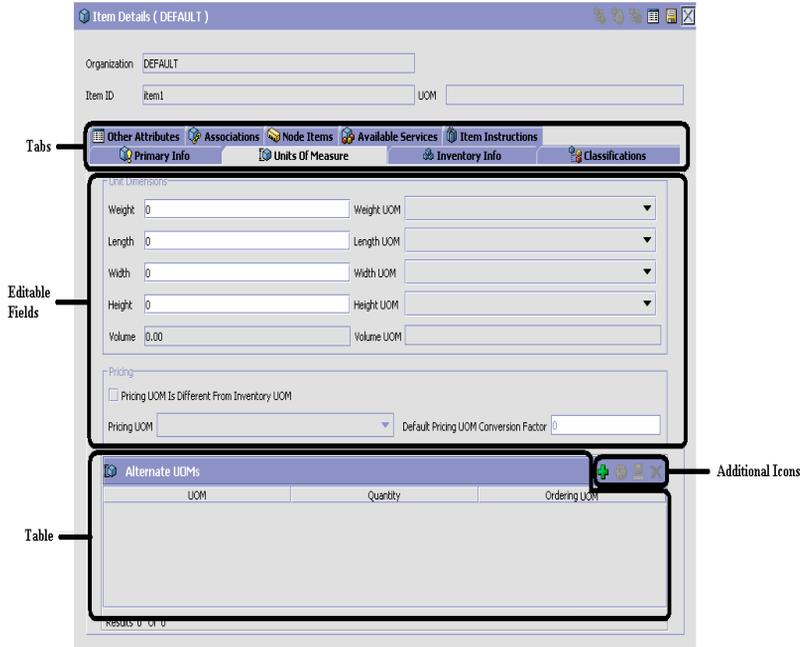
UOM Code	UOM Description
CM	Centimeter
FEET	Feet
IN	Inch
KM	Kilometer
METER	Meter
MILE	Mile

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.

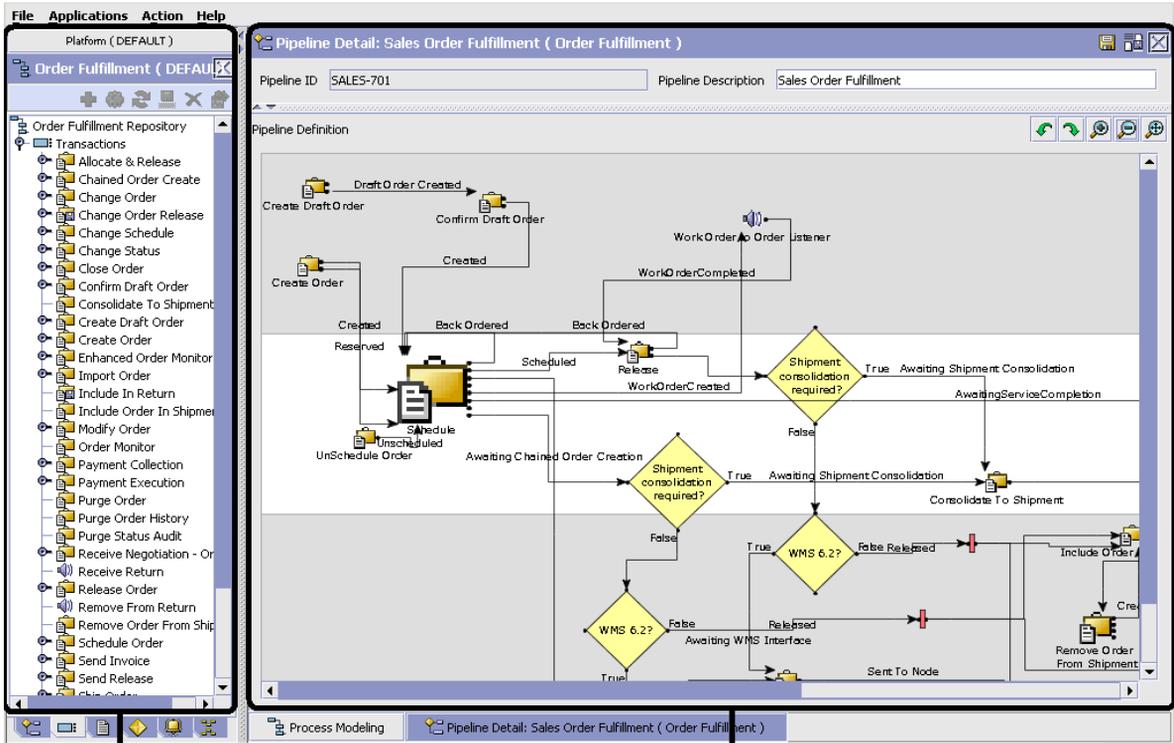
Figure 2–10 Details Window Example



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



Pallet

Graphical Work Area

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

2.3.1 Using the Configurator'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 Configurator, you should refer to the corresponding section in this guide to set up the particular information.

2.3.2 Viewing the Document Types Associated with an Application

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

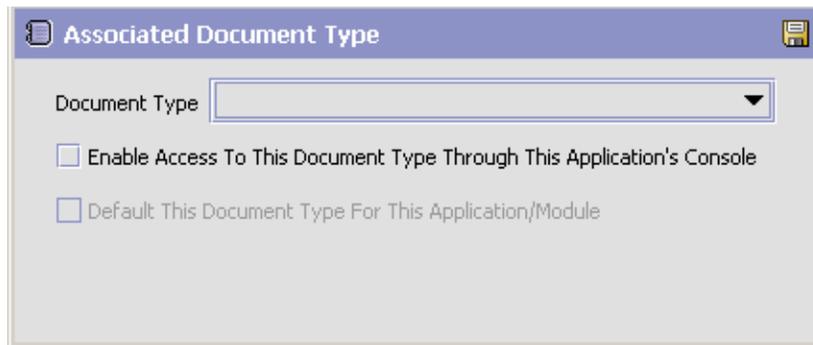
2.3.2.1 Adding a Document Type to an Application

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 Configurator

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

2.3.4 Using Lists and List Filtering

When viewing any list in the Configurator, 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

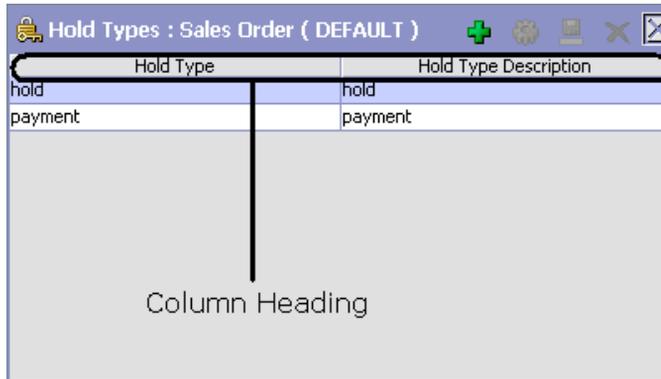


Figure 2–15 Table Filter Editor Window Example

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

Figure 2–16 Calendar Icon example



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

Figure 2–17 Time Field example



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

2.3.6 Using On-Line Help

You can access the 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 Catalog Components

You can define the common codes used throughout the Catalog Management application.

You can use the Catalog Configurations branch for:

- [Defining Item Statuses](#)
- [Defining Additional Attributes](#)
- [Defining Item Instruction Codes](#)
- [Defining Item Instruction Types](#)
- [Defining Service Skills](#)
- [Defining Service Complexity Levels](#)
- [Defining Catalog Rules](#)
- [Defining Assets](#)

3.1 Defining Item Statuses

You can define common codes for item statuses used when setting up an item.

Following are the Selling and Fulfillment Foundation default item statuses:

- 2000 - Held (Items in this status are not available for some operations within Selling and Fulfillment Foundation. For example, an item in this status does not appear within item search screens in the Order Console.)
- 3000 - Published

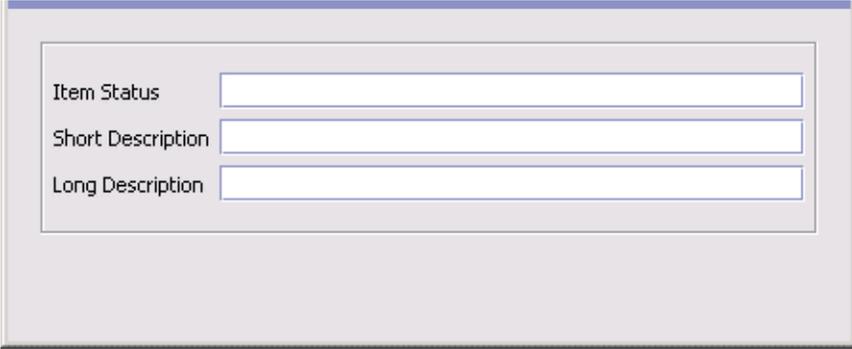
You can use the Item Statuses branch for:

- [Creating an Item Status](#)
- [Modifying an Item Status](#)
- [Deleting an Item Status](#)

3.1.1 Creating an Item Status

To create an item status:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Statuses. The Item Statuses window displays in the work area.
2. Choose . The Item Status Details pop-up window displays.



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

3. In Item Status, enter the value you want to use for the item status.
4. In Short Description, enter a brief description of the item status.
5. In Long Description, enter a more detailed description of the item status.
6. Choose .

3.1.2 Modifying an Item Status

To modify an item status:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Statuses. The Item Statuses window displays in the work area.
2. Select the applicable item status and choose . The Item Status Details pop-up window displays.
3. In Short Description, enter a brief description of the item status.
4. In Long Description, enter a more detailed description of the item status.
5. Choose .

3.1.3 Deleting an Item Status

To delete an item status:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Statuses. The Item Statuses window displays in the work area.
2. Select the applicable item status and choose .

Note: You cannot delete the default item statuses provided in Selling and Fulfillment Foundation.

3.2 Defining Additional Attributes

You can define common codes for additional attributes used when setting up an item. This allows you to store any additional information about your items that is not captured by the defaults in Selling and Fulfillment Foundation. These attributes appear as fields in the item details screen.

Following is the Selling and Fulfillment Foundation default additional attribute:

- REFERENCE

You can use the Additional Attributes branch for:

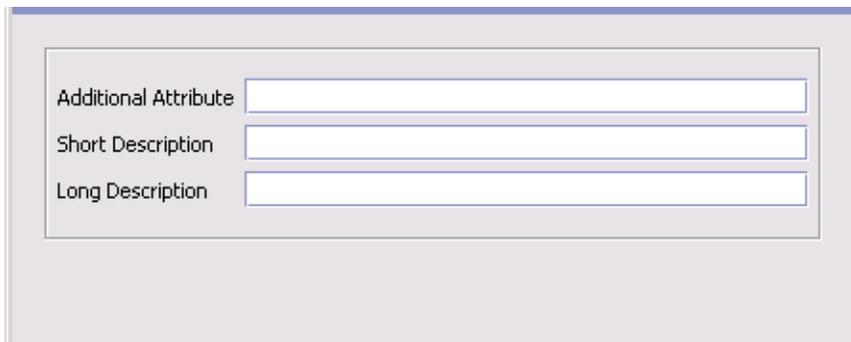
- [Creating an Additional Attribute](#)

- [Modifying an Additional Attribute](#)
- [Deleting an Additional Attribute](#)

3.2.1 Creating an Additional Attribute

To create an additional attribute:

1. From the tree in the application rules side panel, choose Catalog Configurations > Additional Attributes. The Item Additional Attributes window displays in the work area.
2. Choose . The Additional Attribute Details pop-up window displays.



The screenshot shows a pop-up window titled 'Additional Attribute Details'. It contains three text input fields with labels to their left: 'Additional Attribute', 'Short Description', and 'Long Description'. Each field is empty and has a light blue border.

3. In Additional Attribute, enter the value you want to use for the additional attribute.

Important: Do not put a space or special characters in the Additional Attribute.

4. In Short Description, enter a brief description of the additional attribute.
5. In Long Description, enter a more detailed description of the additional attribute.
6. Choose .

3.2.2 Modifying an Additional Attribute

To modify an additional attribute:

1. From the tree in the application rules side panel, choose Catalog Configurations > Additional Attributes. The Item Additional Attributes window displays in the work area.
2. Select the applicable additional attribute and choose . The Additional Attribute Details pop-up window displays.
3. In Short Description, enter a brief description of the additional attribute.
4. In Long Description, enter a more detailed description of the additional attribute.
5. Choose .

3.2.3 Deleting an Additional Attribute

To delete an additional attribute:

1. From the tree in the application rules side panel, choose Catalog Configurations > Additional Attributes. The Item Additional Attributes window displays in the work area.
2. Select the applicable additional attribute and choose .

Note: You cannot delete default additional attributes.

3.3 Defining Item Instruction Codes

You can define codes that describe any special instructions that may be associated with an item. For example, you may want to create a Handle With Care item instruction code to associate with all of your fragile items.

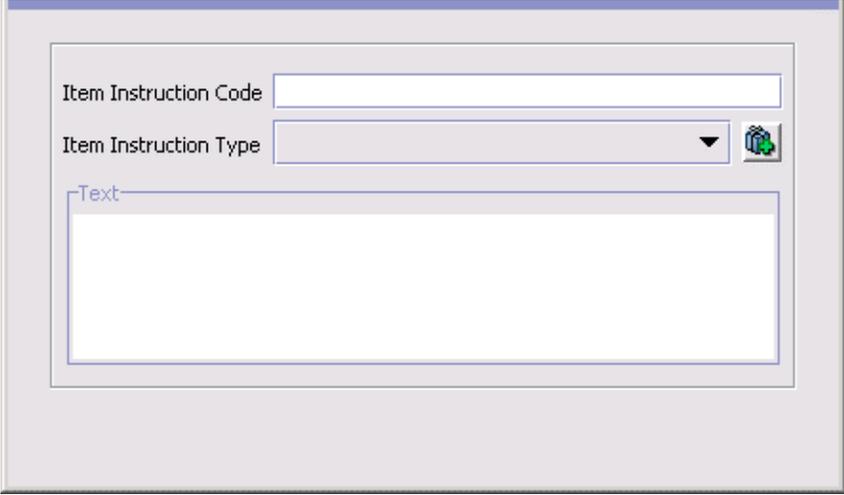
You can use the Item Instruction Codes branch for:

- [Creating an Item Instruction Code](#)
- [Modifying an Item Instruction Code](#)
- [Deleting an Item Instruction Code](#)

3.3.1 Creating an Item Instruction Code

To create an item instruction code:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Instruction Codes. The Item Instruction Codes window displays in the work area.
2. Choose . The Item Instruction Codes pop-up window displays.



The screenshot shows a pop-up window for defining item instruction codes. It features three input fields:

- Item Instruction Code:** A text input field for entering the name of the instruction code.
- Item Instruction Type:** A dropdown menu for selecting the instruction type, accompanied by a plus icon.
- Text:** A large text area for entering special instructions.

3. In Item Instruction Code, enter the name of the item instruction code.
4. From Item Instruction Type, select the instruction type that defines the kind of item instruction you are configuring. For example, Handling or Picking. For more information about configuring item instruction types, see [Section 3.4, "Defining Item Instruction Types"](#).
5. In Text, enter any special instructions needed to further explain the item instruction code.
6. Choose .

3.3.2 Modifying an Item Instruction Code

To modify an item instruction code:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Instruction Codes. The Item Instruction Codes window displays in the work area.
2. Select the applicable item instruction code and choose . The Item Instruction Codes pop-up window displays.
3. From Item Instruction Type, select the instruction type that defines the kind of item instruction you are configuring. For example, Handling or Picking. For more information about configuring item instruction types, see [Section 3.4, "Defining Item Instruction Types"](#).
4. In Text, enter any special instructions needed to further explain the item instruction code.
5. Choose .

3.3.3 Deleting an Item Instruction Code

To delete an item instruction code:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Instruction Codes. The Item Instruction Codes window displays in the work area.
2. Select the applicable item instruction code and choose .

3.4 Defining Item Instruction Types

You can define common codes that can be associated with an item instruction code to further describe what kind of instruction it is. For example, you can create a Pick instruction type and a Handling instruction type. The Pick instruction type can be associated with all item instruction codes that pertain to the item picking process and the Handling instruction type can be associated with all item instruction codes relevant to the handling of items. For more information about configuring item instruction codes, see [Section 3.3, "Defining Item Instruction Codes"](#).

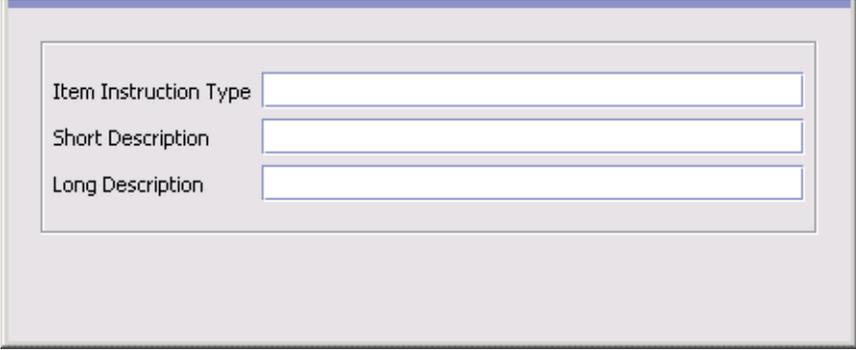
You can use the Item Instruction Types branch for:

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

3.4.1 Creating an Item Instruction Type

To create an item instruction type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Instruction Types. The Item Instruction Types window displays in the work area.
2. Choose . The Item Instruction Type Details pop-up window displays.



Item Instruction Type	<input type="text"/>
Short Description	<input type="text"/>
Long Description	<input type="text"/>

3. In Item Instruction Type, enter the value you want to use for the item instruction type.
4. In Short Description, enter a brief description of the item instruction type.
5. In Long Description, enter a more detailed description of the item instruction type.
6. Choose .

3.4.2 Modifying an Item Instruction Type

To modify an item instruction type:

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

3.4.3 Deleting an Item Instruction Type

To delete an item instruction type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Instruction Types. The Item Instruction Types window displays in the work area.
2. Select the applicable item type and choose .

3.5 Defining Service Skills

You can define services skills when setting up provided service items and delivery service items. You can also associate service skills to a service association for an item.

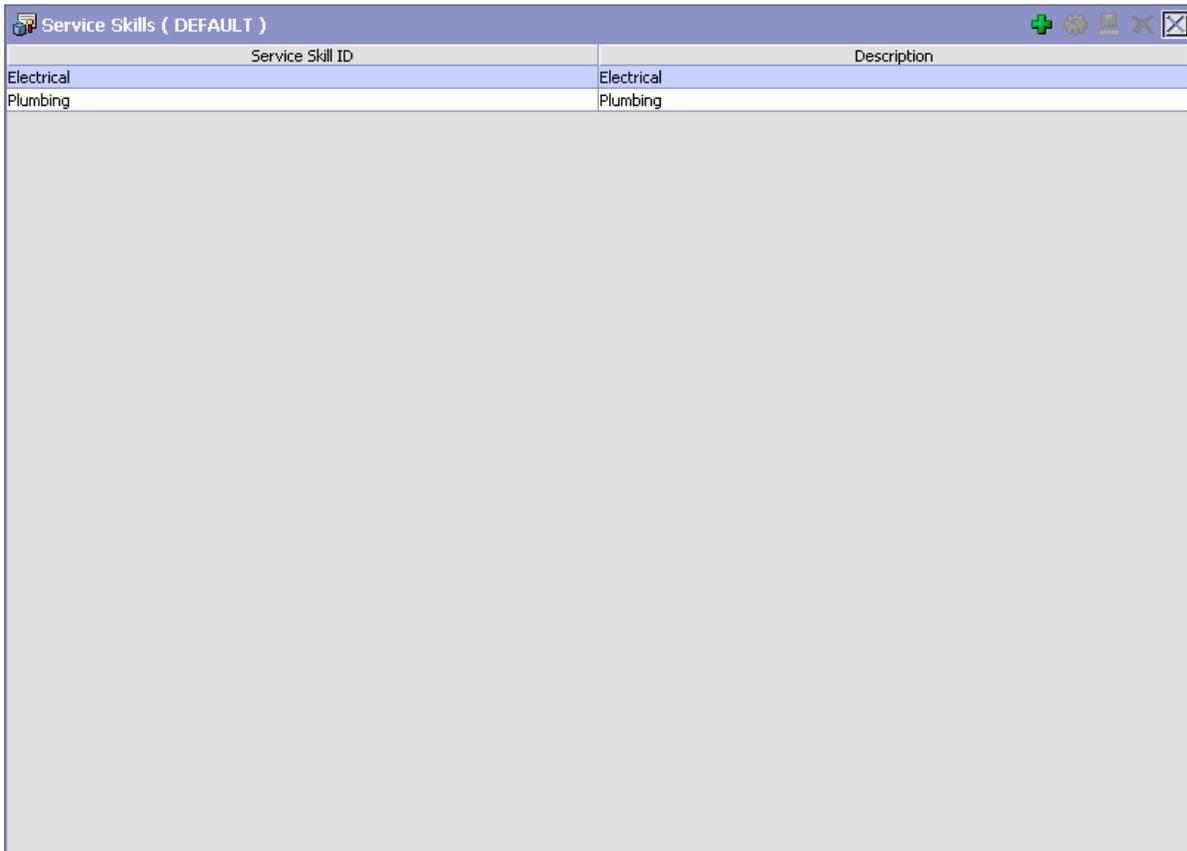
You can use the Service Skills branch for:

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

3.5.1 Creating a Service Skill

To create a service skill:

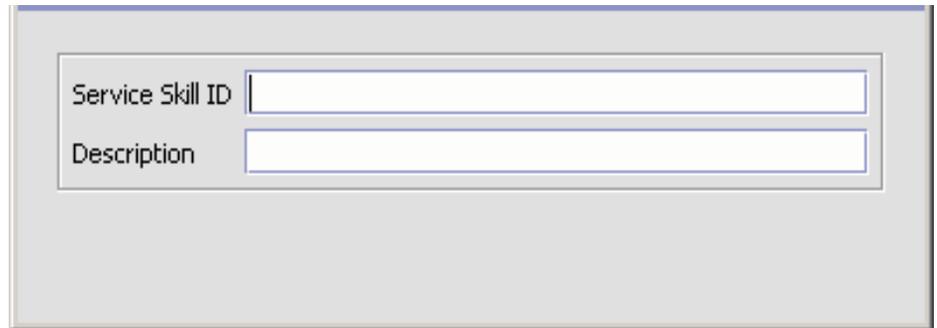
1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Catalog Configurations > Service Skills. The Service Skills window displays in the work area.



The screenshot shows a window titled "Service Skills (DEFAULT)" with a table containing two rows of data. The table has two columns: "Service Skill ID" and "Description".

Service Skill ID	Description
Electrical	Electrical
Plumbing	Plumbing

3. Choose . The Service Skill Details pop-up window displays.



The screenshot shows a dialog box with two text input fields. The first field is labeled "Service Skill ID" and the second is labeled "Description". Both fields are currently empty.

4. In Service Skill ID, enter the value you want to use for the Service Skill.
5. In Description, enter a description of the service skill.
6. Choose .

3.5.2 Modifying a Service Skill

To modify a service skill:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Catalog Configurations > Service Skills. The Service Skills window displays in the work area.
3. Select the applicable service skill and choose . The Service Skill Details pop-up window displays.
4. In Description, enter a description of the service skill.
5. Choose .

3.5.3 Deleting a Service Skill

To delete a service skill:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Catalog Configurations > Service Skills. The Service Skills window displays in the work area.
3. Select the applicable service skill and choose .

3.6 Defining Service Complexity Levels

You can define services complexity levels when setting up provided service items and delivery service items. You can also assign service complexity levels to a service type for an item, which in turn is used to calculate the service complexity level on the work order. See [Section 5.3, "Defining Service Types"](#) or [Section 6.3, "Defining Provided Service Types"](#) for more information.

You can use the Service Complexity Levels branch for:

- [Creating a Service Complexity Level](#)
- [Modifying a Service Complexity Level](#)
- [Deleting a Service Complexity Level](#)

3.6.1 Creating a Service Complexity Level

To create a service complexity level:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Catalog Configurations > Service Complexity Levels. The Service Complexity Level window displays in the work area.
3. Choose . The Service Complexity Level Details pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 3–1](#) for field value descriptions.

The screenshot shows a pop-up window with a light gray background. Inside, there are three text input fields stacked vertically. The first field is labeled 'Service Complexity Level', the second is labeled 'Short Description', and the third is labeled 'Long Description'. Each field has a thin blue border and is currently empty.

Table 3–1 Service Complexity Level Details Pop-up Window

Field	Description
Service Complexity Level	Enter a name for the service complexity level.
Short Description	Enter a short description of the service complexity level.
Long Description	Enter a long description of the service complexity level.

5. Enter information in the applicable fields. Refer to [Table 3–1](#) for more information.
6. Choose .

3.6.2 Modifying a Service Complexity Level

To modify a service complexity level:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Catalog Configurations > Service Complexity Levels. The Service Complexity Level window displays in the work area.
3. Select the applicable service complexity level and choose . The Service Complexity Level Details pop-up window displays.

4. Edit information in the applicable fields. Refer to [Table 3–1](#) for more information.
5. Choose .

3.6.3 Deleting a Service Complexity Level

To delete a service complexity level:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Catalog Configurations > Service Complexity Levels. The Service Levels window displays in the work area.
3. Select the applicable service complexity level and choose .

3.7 Defining Item Alias Types

You can define common codes for item alias types used when configuring product items, provided services, and delivery services. Aliases appear as fields in the item details screen.

Following is the Selling and Fulfillment Foundation default alias type:

- UPC Code

You can use the Alias Types branch for:

- [Creating an Item Alias Type](#)
- [Modifying an Item Alias Type](#)
- [Deleting an Item Alias Type](#)

3.7.1 Creating an Item Alias Type

To create an item alias type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Alias Types. The Item Aliases window displays in the work area.
2. Choose . The Alias Type Details pop-up window displays.

The screenshot shows a configuration window with three text input fields. The first field is labeled 'Item Alias Type', the second is 'Short Description', and the third is 'Long Description'. Each field is empty and has a light blue border.

3. In Item Alias Type, enter the value you want to use for the alias type.

Important: The Item Alias Type cannot begin with a number or have a space or special characters in it.

4. In Short Description, enter a brief description of the item alias type.
5. In Long Description, enter a more detailed description of the item alias type.
6. Choose .

3.7.2 Modifying an Item Alias Type

To modify an item alias type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Alias Types. The Item Aliases window displays in the work area.
2. Select the applicable alias type and choose . The Alias Type Details pop-up window displays.
3. In Short Description, enter a brief description of the item alias type.
4. In Long Description, enter a more detailed description of the item alias type.
5. Choose .

3.7.3 Deleting an Item Alias Type

To delete an item alias type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Alias Types. The Item Aliases window displays in the work area.
2. Select the applicable item alias type and choose .

Note: You cannot delete default item alias types.

3.8 Defining Item Association Types

You can define common codes for item association types used when configuring product items, provided services, and delivery services.

Following are the Selling and Fulfillment Foundation default association types:

- CrossSell
- Substitutions
- UpSell

You can use the Association Types branch for:

- [Creating an Item Association Type](#)
- [Modifying an Item Association Type](#)
- [Deleting an Item Association Type](#)

3.8.1 Creating an Item Association Type

To create an item association type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Association Types. The Item Association Types window displays in the work area.
2. Choose . The Item Association Type Details pop-up window displays.

The screenshot shows a configuration window with three text input fields. The first field is labeled 'Item Association Type', the second is 'Short Description', and the third is 'Long Description'. Each field is empty and has a light blue border.

3. In Item Association Type, enter the value you want to use for the association type.

Important: Do not put a space or special characters in the Item Association Type.

4. In Short Description, enter a brief description of the item association type.
5. In Long Description, enter a more detailed description of the item association type.
6. Choose .

3.8.2 Modifying an Item Association Type

To modify an item association type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Association Types. The Item Association Types window displays in the work area.
2. Select the applicable association type and choose . The Item Association Type Details pop-up window displays.
3. In Short Description, enter a brief description of the item association type.

4. In Long Description, enter a more detailed description of the item association type.
5. Choose .

3.8.3 Deleting an Item Association Type

To delete an item association type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Association Types. The Item Association Types window displays in the work area.
2. Select the applicable item association type and choose .

Note: You cannot delete default item association types.

3.9 Defining Item Exclusion Codes

You can define common codes for item exclusion codes used when configuring product items, provided services, and delivery services. An item exclusion code may be set up if your business practices involve the regulation of item shipments for certain countries. For example, if there are certain countries that you ship to that do not allow items with certain hazardous materials to be shipped to them, you may create an item exclusion code that you can associate with any of your items that may contain the certain hazardous material.

Note: Selling and Fulfillment Foundation does not by default provide logic to handle item exclusion codes. This functionality is provided to enable integration of Selling and Fulfillment Foundation with external systems that can handle item exclusion codes.

Following is the Selling and Fulfillment Foundation default item exclusion code:

- DEFAULT

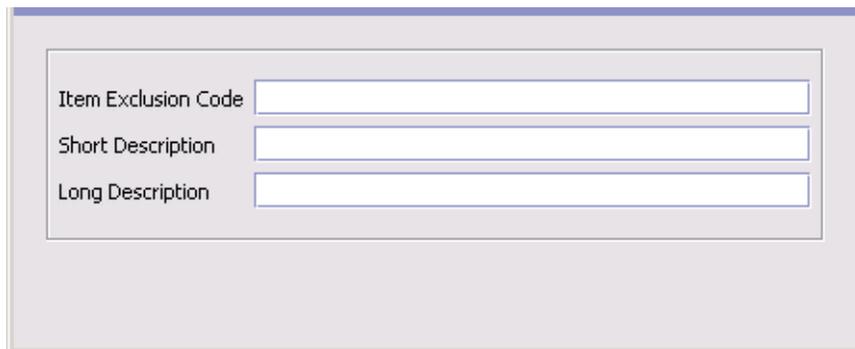
You can use the Item Exclusion Codes branch for:

- [Creating an Item Exclusion Code](#)
- [Modifying an Item Exclusion Code](#)
- [Deleting an Item Exclusion Code](#)

3.9.1 Creating an Item Exclusion Code

To create an item exclusion code:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Exclusion Codes. The Item Exclusion Codes window displays in the work area.
2. Choose . The Item Exclusion Code Details pop-up window displays.



The screenshot shows a pop-up window titled 'Item Exclusion Code Details'. It contains three text input fields arranged vertically. The first field is labeled 'Item Exclusion Code', the second is labeled 'Short Description', and the third is labeled 'Long Description'. Each field is currently empty.

3. In Item Exclusion Code, enter the value you want to use for the item exclusion code.
4. In Short Description, enter a brief description of the item exclusion code.
5. In Long Description, enter a more detailed description of the item exclusion code.
6. Choose .

3.9.2 Modifying an Item Exclusion Code

To modify an item exclusion code:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Exclusion Codes. The Item Exclusion Codes window displays in the work area.
2. Select the applicable item exclusion code and choose . The Item Exclusion Code Details pop-up window displays.
3. In Short Description, enter a brief description of the item exclusion code.
4. In Long Description, enter a more detailed description of the item exclusion code.
5. Choose .

3.9.3 Deleting an Item Exclusion Code

To delete an item exclusion code:

1. From the tree in the application rules side panel, choose Catalog Configurations > Item Exclusion Codes. The Item Exclusion Codes window displays in the work area.
2. Select the applicable item exclusion code and choose .

Note: You cannot delete default item exclusion codes.

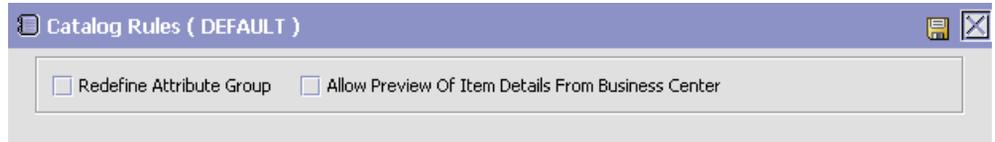
3.10 Defining Catalog Rules

You can define a catalog rule that allows product administrators to redefine attribute groups when assigning attribute groups to categories and classifications. Attribute groups are first defined when you set up master data for the attribute domain; however, you may want to redefine attribute groups later when assigning them to categories and classifications. The Redefine Attribute Groups option allows you to redefine attribute groups.

In addition, you can define a catalog rule that allows users to preview item details from the Business Center application. The item details may be a screen on the user's internal or external Web site. This allows users to dynamically view changes that they may have just made to an item.

To allow users to redefine attribute groups and to preview item details from the Business Center application:

1. From the tree in the application rules side panel, choose Catalog Configurations > Catalog Rules. The Catalog Rules window displays in the work area.



2. Select Redefine Attribute Group to specify that you want to allow users to redefine attribute groups when assigning them to categories in the master catalog.
3. Select Allow Preview Of Item Details From Business Center to specify that you want to allow users to preview item details from the Business Center application.
4. Choose .

3.11 Defining Assets

You define assets by creating the following:

- Common codes for assets types
- Content locations for assets

Assets are electronic media objects that you associate with items, categories, attributes, and allowed attribute values. Asset types can include images, video, audio, and text files.

In addition to defining common codes for asset types, you must specify content locations for assets. Typically, assets are digitally-based objects that reside in external systems. For example, you can assign an image file to items that are sold from an online catalog so that online customers can view pictures of items. In this case, you must provide a content location for the image file. The content location is the path that identifies the location of the image file.

If a user defines hub and enterprise asset types or content locations, the combination of the two are used at the enterprise level.

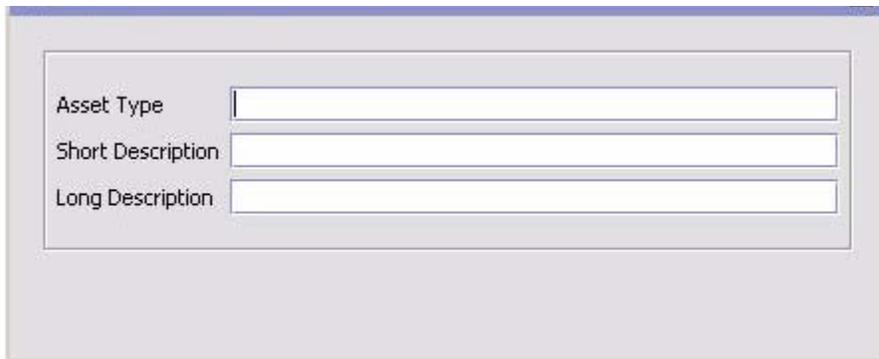
You can use the Asset branch for:

- [Creating an Asset Type](#)
- [Modifying an Asset Type](#)
- [Deleting an Asset Type](#)
- [Creating an Asset Content Location](#)
- [Modifying an Asset Content Location](#)
- [Deleting an Asset Content Location](#)

3.11.1 Creating an Asset Type

To create an asset type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Asset > Asset Types. The Asset Types window displays in the work area.
2. Choose . The Asset Type Details pop-up window displays.



The screenshot shows a pop-up window titled "Asset Type Details" with a light gray background. Inside the window, there is a white rectangular area containing three text input fields. The first field is labeled "Asset Type", the second is labeled "Short Description", and the third is labeled "Long Description". Each label is positioned to the left of its corresponding text box, which has a thin border and a vertical cursor on the left side.

3. In Asset Type, enter the value you want to use for the asset type.

Note: If you are defining an asset type for an image and want to allow the Business Center administrator to preview the corresponding image from the Business Center, ensure that the value for the asset type starts with "Image". For example, if you are defining an asset type for small images, you could specify the asset type's value as "Imagesmallpicture" or "Image_small".

4. In Short Description, enter a brief description of the asset type.
5. In Long Description, enter a more detailed description of the asset type.
6. Choose .

3.11.2 Modifying an Asset Type

To modify an asset type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Asset > Asset Types. The Asset Types window displays in the work area.
2. Select the applicable asset type and choose . The Asset Type Details pop-up window displays.
3. In Short Description, enter a brief description of the asset type.
4. In Long Description, enter a more detailed description of the asset type.
5. Choose .

3.11.3 Deleting an Asset Type

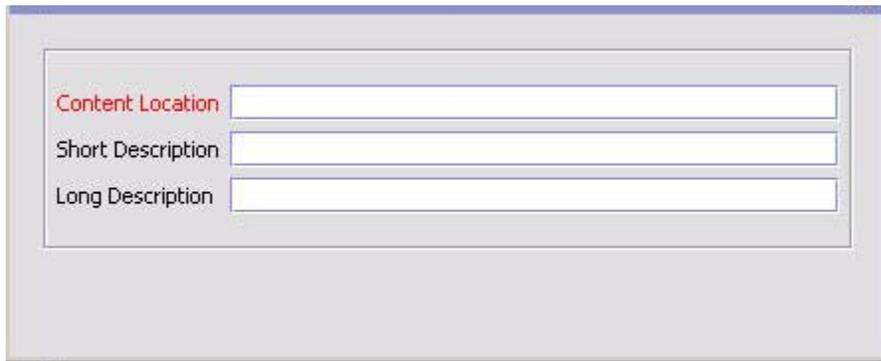
To delete an asset type:

1. From the tree in the application rules side panel, choose Catalog Configurations > Asset > Asset Types. The Asset Types window displays in the work area.
2. Select the applicable asset type and choose .

3.11.4 Creating an Asset Content Location

To create an asset content location:

1. From the tree in the application rules side panel, choose Catalog Configurations > Asset > Content Locations. The Content Locations window displays in the work area.
2. Choose . The Content Location Details pop-up window displays.



The image shows a pop-up window titled 'Content Location Details'. It contains three text input fields stacked vertically. The first field is labeled 'Content Location' in red text. The second field is labeled 'Short Description'. The third field is labeled 'Long Description'. Each field has a light gray border and a white background.

3. In Content Location, enter the value you want to use for the content location.
4. In Short Description, enter a brief description of the content location.
5. In Long Description, enter a more detailed description of the content location.
6. Choose .

3.11.5 Modifying an Asset Content Location

To modify an asset content location:

1. From the tree in the application rules side panel, choose Catalog Configurations > Asset > Content Locations. The Content Locations window displays in the work area.
2. Select the applicable content location and choose . The Content Locations Details pop-up window displays.
3. In Short Description, enter a brief description of the content location.

4. In Long Description, enter a more detailed description of the content location.
5. Choose .

3.11.6 Deleting an Asset Content Location

To delete an asset content location:

1. From the tree in the application rules side panel, choose Catalog Configurations > Asset > Content Locations. The Content Locations window displays in the work area.
2. Select the applicable content location and choose .

4

Configuring Products

A product item is a physical unit that can be ordered, shipped, and returned. A product item is uniquely defined by its item ID and unit of measure. You can configure items, units of measure, master catalogs, categories, classifications, and additional item attributes. For more information about configuring product item inventory rules, see the *Sterling Global Inventory Visibility: Configuration Guide*. For more information about configuring product item sourcing rules, see the *Sterling Distributed Order Management: Configuration Guide*.

You can use the Products branch for:

- [Defining Product Items](#)
- [Defining Item Attribute Overrides at the Node Level](#)
- [Defining Master Units of Measure](#)
- [Defining Master Catalogs](#)
- [Defining Categories](#)
- [Defining Classifications](#)

4.1 Defining Product Items

Items are used to create product items that can be used in Selling and Fulfillment Foundation, as well as define their unique attributes.

You can use the Items branch for:

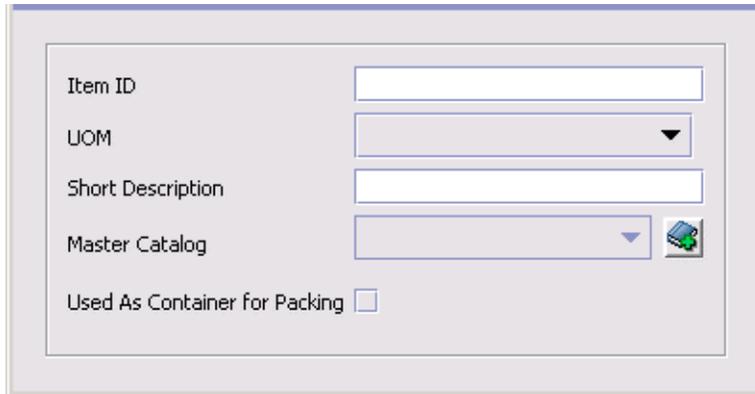
- [Creating a Product Item](#)
- [Modifying a Product Item](#)
- [Recomputing an Item's Computed Unit Cost](#)

- [Using the Condition Builder](#)

4.1.1 Creating a Product Item

To create an item:

1. From the tree in the application rules side panel, choose Products > Items. The Product Item Search window displays in the work area.
2. Choose . The Create Item pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 4-1](#) for field value descriptions.
4. Choose .



The screenshot shows a 'Create Item' pop-up window with the following fields:

Item ID	<input type="text"/>
UOM	<input type="text" value="▼"/>
Short Description	<input type="text"/>
Master Catalog	<input type="text" value="▼"/> 
Used As Container for Packing	<input type="checkbox"/>

Table 4–1 Create Item Pop-Up Window

Field	Description
Item ID	Enter the item ID.
UOM	Select the quantity unit of measure you want to associate with the item. Note: The unit of measure makes the item ID unique. For example, if you sell the same item with the same item ID from a node in the United States and a node in Canada, and the United States uses pounds for the unit of measure and Canada uses kilograms, the two different units of measure associated with the item ID makes two unique items in the system.
Short Description	Enter a brief description of the item.
Master Catalog	Select the master catalog you want this item to be part of. A master catalog is the exhaustive list of an organization's items. For more information about configuring master catalogs, see Section 4.4, "Defining Master Catalogs" .
Used as a Container for Packing	Select to indicate this Item is a container. See Section 4.1.2.14, "Defining a Product Item's Container Attributes" .

Note: No Graphical User Interface is provided for creating or modifying Hazmat data.

4.1.2 Modifying a Product Item

To modify an item:

1. From the menu bar, choose Applications > Catalog Management > Items. The Item Search window displays.
2. Enter applicable search criteria and choose . The Item List displays.
3. Select the applicable item and choose . The Item Details window displays.

Use the Item Details window for:

- [Defining a Product Item's Primary Information](#)
- [Defining a Product Item's Units of Measure](#)

- [Defining a Product Item's Sourcing and Transportation Info](#)
- [Defining a Product Item's Inventory Information](#)
- [Defining a Product Item's Classifications](#)
- [Defining a Product Item's Additional Attributes](#)
- [Defining Product Item Associations](#)
- [Defining a Product Item's Node Item Attribute Overrides](#)
- [Defining a Product Item's Associated Service Items](#)
- [Defining a Product Item's Instructions](#)
- [Defining a Product Item's Components](#)
- [Defining a Product Item's Container Attributes](#)
- [Defining a Product Item's Extended Attributes](#)

4.1.2.1 Defining a Product Item's Primary Information

A product item's primary information provides general information about the item.

Some of these attributes can also be specified at the classification level. For more information about viewing inheritable attributes, see [Section 4.1.2.5, "Viewing Inheritable Attributes"](#).

For more information about defining common item attributes, see [Section 4.6.3.4, "Defining Common Item Attributes for a Classification Value"](#).

To set up an item's primary information:

1. In the Item Details window, choose the Primary Info tab.
2. Enter information in the applicable fields. Refer to [Table 4–2](#) for field value descriptions.

Table 4–2 Primary Info Tab

Field	Description
Master Catalog	Select the master catalog in which you want to include the item.
Is Item With Variations	Check this box if the item is an item with variations. This will cause the Children Items tab to appear. Note: Adding an item with variations to a bundle causes an error to be thrown during the create order or change order process.
Short Description	Enter a brief description of the item.

Table 4–2 Primary Info Tab

Field	Description
Kit Code	<p>If the item is a kit item, select whether it is a physical kit, a bundle, or a dynamic physical kit.</p> <p>A physical kit is maintained as a single item.</p> <p>A dynamic physical kit contains individual kit items that may require value-added services to be performed at the node. If you select dynamic physical kit, the Other Services tab is enabled on the Available Services tab. From this tab kitting and de-kitting services can be selected for the creation of appropriate work orders for the kit creation.</p> <p>A bundle can contain items, services, and other bundles. If you select bundle as the kit code, the Components tab is enabled. In the Components tab you can enter product details. When "Enable Logical Kit" is checked, logical kit is listed in the drop-down list. Otherwise only physical kit, dynamic physical kit, and bundle appear in the list.</p> <p>Note: If you select a kit code for an item an additional Components tab displays in the Item Details window.</p> <p>Note: If you change the kit code for an item, check the component list of the kit to make sure that the change is correct.</p>
Description	Enter a more detailed description of the item.
GTIN	Enter the item's Global Trade Item Number (GTIN). GTIN is the evolving standard for product identification. Selling and Fulfillment Foundation uses GTIN to cross-reference between item definitions of different catalog organizations. For more information about defining catalog organizations, see the <i>Selling and Fulfillment Foundation: Application Platform Configuration Guide</i> .
GTIN Manufacturer Prefix Length	Manufacturer's prefix length specified in the GTIN.
Status	<p>Select Held to make the item unavailable. Such items are still able to be processed by the Selling and Fulfillment Foundation APIs.</p> <p>Choose Published to make the item available to the APIs through the UI.</p>

Table 4–2 Primary Info Tab

Field	Description
Cost Currency	The currency used for the cost of the item, such as Euro or Dollar.
Minimum Order Quantity	Enter a minimum quantity that an order can be made for against the item. Note: This field has no logic associated with it and can be configured as per your business practices. However, the system does not allow you to enter a minimum quantity that is greater than the maximum quantity.
Maximum Order Quantity	Enter a maximum quantity that an order can be made for against the item. Note: This field has no logic associated with it and can be configured as per your business practices. However, the system does not allow you to enter a maximum quantity that is less than the minimum quantity.
Unit Cost	Enter the cost of the item. All cycle counts are based on the unit cost of the item.
Computed Cost	The computed cost of the item.
Size Code	Enter the size code of the item.
Color Code	Enter the color code of the item.
Taxable	Check this box if the item is taxed.
Can Be Added To Work Order As Service Tool	Check this box to allow the item to be used as a service tool on a work order. A service tool can still serve as a regular product item whose inventory can be adjusted and tracked within Selling and Fulfillment Foundation. Additionally, it can be associated with a work order as a tool that may be used by a resource to run the work order's services.
Source	
Manufacturer's Name	Enter the name of the manufacturer who made the item.
Manufacturer's Item	Enter the identifier that the manufacturer uses to identify the item.
Manufacturer's Item Description	Enter the description the manufacturer uses for the item.
Country of Origin	Click  and select the country code.

Table 4–2 Primary Info Tab

Field	Description
Reverse Logistics	
Returnable	Select "YES" if the buyer can return the item.
Return Window Days	Enter the number of days from receipt of the item that the buyer has to return the item.
Return Shipping Label not required	Choose this option if you do not require a return shipping label. You can choose this option only if you select "YES" from the Returnable drop-down list.
Return Shipping Label required at SKU level	Choose this option if you require return labels for each unit of the item packaged in the container. You can choose this option only if you select "YES" from the Returnable drop-down list.
Return Shipping Label required at package level	Choose this option if you require return labels for each container in which the item is packed. You can choose this option only if you select "YES" from the Returnable drop-down list.
Credit Without Receipt	Check this box if the Buyer can receive credit for the item before returning it.
Extended Description	You can enter a detailed description about the item including any notes or handling instructions. Choose the Localize button to enter the description in the multiple languages of the locales you may have configured. Note: All locale configuration should be completed before attempting to enter an extended item description. For more information about configuring locales, see the <i>Selling and Fulfillment Foundation: Application Platform Configuration Guide</i> .

4.1.2.2 Defining a Product Item’s Units of Measure

You can provide an item’s weight and dimensions for shipping and storage purposes. You can also configure any alternate pricing and ordering units of measure an item may have.

To set up an item’s measurements:

1. In the Item Details window, choose the Units Of Measure tab.
2. Enter information in the applicable fields. Refer to [Table 4–3](#) for field value descriptions.

Table 4–3 Units Of Measure Tab

Field	Description
Weight	Enter the item’s numerical weight and select its unit of measure (for example, lbs.) from Weight UOM.
Length	Enter the item’s numerical length and select its unit of measure (for example, inches) from Length UOM.
Width	Enter the item’s numerical width and select its unit of measure (for example, inches) from Width UOM.
Height	Enter the item’s numerical height and select its unit of measure (for example, inches) from Height UOM.
Pricing	

Table 4–3 Units Of Measure Tab

Field	Description
Pricing UOM Different Is From Inventory UOM	Select Pricing UOM Different Is From Inventory UOM, if this item uses a different unit of measure for pricing calculations than the unit of measure associated with the actual inventory product item. For example, you are creating a product item called Apples. Apples are inventoried and sold as a unit of measure called Cases. However, the pricing is determined by the weight of each case. In this scenario, you would assign the product item Apples the inventory unit of measure Cases and the pricing unit of measure Weight.
Pricing UOM	If you selected Pricing UOM Different Is From Inventory UOM, select the unit of measure you want to use to price this item.
Default Pricing UOM Conversion Factor	Enter the conversion rate from the inventory unit of measure to the pricing unit of measure. For example, a product item has a inventory unit of measure of EACH and a pricing unit of measure of CASE. 4 EACH make up 1 CASE. In this scenario, you would enter '4' as the conversion factor.
Alternate UOMs	A list of alternate units of measure used by the item ID/unit of measure combination you are defining. For more information about alternate ordering units of measure, see Section 4.1.2.2.1, "Adding an Alternate Unit of Measure to a Product Item" .
UOM	The unit of measure.
Quantity	The conversion rate from the inventory unit of measure to the alternate unit of measure.
Ordering UOM	Indicates that the unit of measure can be used in the selling environment.

Note: Volume specified for cases or pallets in the Alternate UOM screen is considered for deciding whether over packing is required during the pack process. If the volume of the case or pallet is less than the volume defined in the Pack Strategy screen, the container gets over packed.

You can use the Units Of Measure tab for:

- [Adding an Alternate Unit of Measure to a Product Item](#)
- [Modifying a Product Item's Alternate Unit of Measure](#)
- [Removing a Product Item's Alternate Unit of Measure](#)

4.1.2.2.1 Adding an Alternate Unit of Measure to a Product Item

You may need to create a product item that has multiple ways in which its unique item ID/unit of measure combination can be packaged. In this case you can create alternate ordering units of measure that can be used in the selling environment.

For example, you may create a unique item ID/unit of measure combination for a 6-Pack of Pens. However, in your selling environment you sell this item one of three ways: individually, cases, and pallets. In this scenario you can create three alternate units of measure that can be associated with the 6-Pack inventory unit of measure: 6-Pack, Case, and Pallet.

To add an alternate unit of measure:

1. In the Units of Measure tab, choose  from the Alternate UOMs table. The Alternate UOM Details pop-up window displays.
2. Enter information in the applicable fields. Refer to [Table 4–4](#) for field value descriptions.
3. Choose .

The screenshot shows a configuration window for defining an alternate unit of measure (UOM). It includes the following fields and options:

- Alternate UOM:** A dropdown menu.
- Conversion Quantity to Inventory UOM:** A text input field.
- Positive Package Tolerance Qty:** A text input field.
- Negative Package Tolerance Qty:** A text input field.
- Orders Can Be Placed Using This UOM
- Unit Dimensions:** A section containing four rows:
 - Weight:** Text input and dropdown for Weight UOM.
 - Length:** Text input and dropdown for Length UOM.
 - Width:** Text input and dropdown for Width UOM.
 - Height:** Text input and dropdown for Height UOM.

Table 4–4 *Alternate UOM Details Pop-Up Window*

Field	Description
Alternate UOM	Select the alternate unit of measure you want to use.
Conversion Quantity to Inventory UOM	Enter the conversion quantity from the inventory unit of measure to the alternate unit of measure. For example, you are configuring an alternate unit of measure called DOZEN for a product item with an inventory unit of measure of EACH. 12 EACH make up 1 DOZEN. In this scenario, you would enter '12' as the conversion quantity. Important: This field should be configured at the beginning of an implementation. Do not modify this field once orders exist in the system.
Positive Package Tolerance Qty	This field is not used in this release.
Negative Package Tolerance Qty	This field is not used in this release.

Table 4–4 Alternate UOM Details Pop-Up Window

Field	Description
Orders Can Be Placed Using This UOM	Select Orders Can Be Placed Using This UOM if this unit of measure can be used for the particular item ID/unit of measure combination in a selling environment.
Unit Dimensions	
Weight	Enter the unit of measure's weight.
Length	Enter the unit of measure's length.
Width	Enter the unit of measure's width.
Height	Enter the unit of measure's height.

4.1.2.2.2 Modifying a Product Item's Alternate Unit of Measure

To modify an item's alternate unit of measure:

1. In the Units of Measure tab, select the applicable alternate unit of measure from the Alternate UOMs table and choose . The Alternate UOM Details pop-up window displays.
2. Enter information in the applicable fields. Refer to [Table 4–4](#) for field value descriptions.
3. Choose .

4.1.2.2.3 Removing a Product Item's Alternate Unit of Measure

To remove a product item's alternate unit of measure, select the applicable unit of measure from the Alternate UOMs table and choose .

4.1.2.3 Defining a Product Item's Sourcing and Transportation Info

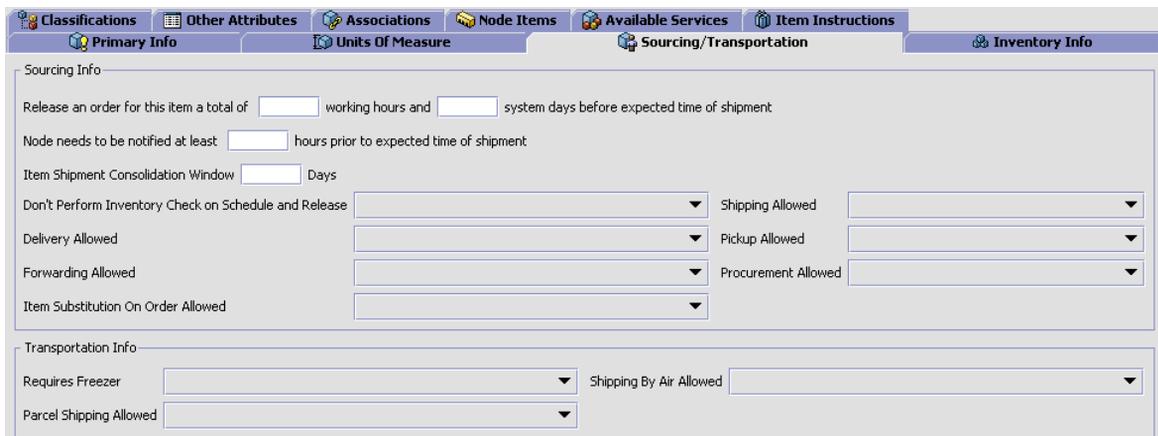
You can configure attributes for the item's sourcing and transportation information that determine how it is monitored and handled during sourcing and transportation.

Some of these attributes can also be specified at the classification level. For more information about viewing inheritable attributes, see [Section 4.1.2.5, "Viewing Inheritable Attributes"](#).

For more information about defining common item attributes, see [Section 4.6.3.4, "Defining Common Item Attributes for a Classification Value"](#).

To set up an item's sourcing and transportation information:

1. In the Item Details window, choose the Sourcing and Transportation Info tab.
2. Enter information in the applicable fields. Refer to [Table 4–5](#) for field value descriptions.
3. Choose .



The screenshot shows the 'Sourcing/Transportation' tab in the Item Details window. The window has a menu bar with 'Classifications', 'Other Attributes', 'Associations', 'Node Items', 'Available Services', and 'Item Instructions'. Below the menu bar are four tabs: 'Primary Info', 'Units Of Measure', 'Sourcing/Transportation' (selected), and 'Inventory Info'. The 'Sourcing Info' section contains the following fields:

- Release an order for this item a total of working hours and system days before expected time of shipment
- Node needs to be notified at least hours prior to expected time of shipment
- Item Shipment Consolidation Window Days
- Don't Perform Inventory Check on Schedule and Release
- Shipping Allowed
- Delivery Allowed
- Pickup Allowed
- Forwarding Allowed
- Procurement Allowed
- Item Substitution On Order Allowed

The 'Transportation Info' section contains the following fields:

- Requires Freezer
- Shipping By Air Allowed
- Parcel Shipping Allowed

Table 4–5 Sourcing and Transportation Info (Item Details)

Field	Description
Sourcing Info	
Release an order for this item a total of <number of hours> working hours and <number of days> system days before expected time of shipment	Enter the total number of working hours and system days an order for this item should be released before its expected to ship.
Node needs to be notified at least <number of hours> hours prior to expected time of shipment	Enter the minimum number of hours a node needs to be notified before the expected time of shipment.
Item Shipment Consolidation Window <number of days> Days	Enter the number of days an item can be delayed after its availability date in order to consolidate with other items for shipment.
Don't Perform Inventory Check on Schedule and Release	Specify whether inventory checks should be performed during the Schedule and Release process.
Shipping Allowed	Specify whether shipping is allowed. Options are: <ul style="list-style-type: none"> • Y - Yes, shipping is allowed • M - Maybe, only if delivery is not possible • N - No, shipping is not allowed
Delivery Allowed	Specify whether delivery is allowed. Options are: <ul style="list-style-type: none"> • Y - Yes, delivery is allowed • M - Maybe, only if something else is being delivered • N - No, delivery is not allowed
Pickup Allowed	Specify whether pickup is allowed.

Table 4–5 Sourcing and Transportation Info (Item Details)

Field	Description
Forwarding Allowed	Specify whether forwarding is allowed.
Procurement Allowed	Specify whether procurement is allowed.
Item Substitution on Order Allowed	Specify whether substitution for the item is allowed.
Transportation Info	
Requires Freezer	Specify whether the items require freezer storage during transportation.
Shipping By Air Allowed	Specify whether the items can be shipped by air.
Parcel Shipping Allowed	Specify whether parcel shipping is allowed for the transfer of items.

4.1.2.4 Defining a Product Item’s Inventory Information

You can configure attributes for an item’s inventory information that determine how it is monitored and handled for the overall inventory picture. The overall inventory picture provides the Application Console user with information about an item’s availability and demand. This information can also be used to determine how long it takes to process the item and, if applicable, how long before the item expires.

You can also determine whether or not an item is tag controlled. A tag number is used to uniquely identify an item in the system regardless of the inventory identification numbers you use (for example, lot number and revision number). Inventory identification numbers are used to differentiate products within inventory, physically and systematically, because these “product instances” have different characteristics. Some common examples of identification numbers are lot number, revision number, and manufacturing batch number.

While an organization may have multiple inventory identification numbers, these identification numbers are not necessarily relevant to every product in the organization’s catalog. For some products, a lot number uniquely defines all characteristics, whereas revision number differentiates another product. The inventory tag number is used to rationalize these different situations. In most cases, the tag number

represents one of the inventory identification numbers such as lot number, batch number, or revision number based on the product.

For example, you have two different items, one called Item A that is uniquely identified by a lot number and one called Item B that is uniquely identified by a revision number. In this case, when an instance of Item A is created with a lot number of 12345, it is also assigned a tag number of 12345. When an instance of Item B is created with a revision number of 98765, it is also assigned a tag number of 98765.

Some of these attributes can also be specified at the classification level. For more information about viewing inheritable attributes, see [Section 4.1.2.5, "Viewing Inheritable Attributes"](#).

For more information about defining common item attributes, see [Section 4.6.3.4, "Defining Common Item Attributes for a Classification Value"](#).

To set up an item's inventory information:

1. In the Item Details window, choose the Inventory Info tab.
2. Enter information in the applicable fields. Refer to [Table 4–6](#) for field value descriptions.

The screenshot shows the 'Inventory Info' tab of the Item Details window. The interface includes the following elements:

- Time Sensitive:**
- Default Expiration Days:** 0
- ATP Rule:** [Dropdown]
- ATP Monitor Rule:** [Dropdown]
- Node Level ATP Monitor Rule:** [Dropdown]
- Default Product Class:** [Dropdown]
- Track FIFO in Inventory:** [Dropdown]
- Lead Days:** [Text Field]
- Future Safety Factor Percentage:** [Text Field]
- Primary Supplier:** [Dropdown]
- Onhand Safety Factor Percentage:** [Text Field]
- Onhand Safety Factor Quantity:** [Text Field]
- Serial Tracked:**
- Serialized:**
- Number Of Secondary Serials:** 0
- Item Based Allocation Allowed:** Yes
- Use Unplanned Inventory:** [Dropdown]
- Tag Controlled:**
 - Not Tag Controlled
 - Sometimes Tag Controlled
 - Always Tag Controlled

Table 4–6 Inventory Info Tab

Field	Description
Time Sensitive	<p>Select Time Sensitive if the item is perishable with a specific ship-by date.</p> <p>If the ship-by date is not provided for a time sensitive item:</p> <ul style="list-style-type: none"> For a positive adjustment the ship by date is calculated as the values of System Date + Default Expiration Days and adds Supply. For a negative adjustments, the system reduces the supply in the order of First Expired First Out (FEFO). If there is not enough available quantity, a new supply record with a negative figure is created for a date 01/01/2500. For example, if FEFO is allowed and no ship by date is entered and the user tries to reduce inventory by 50 when a supply record of 40 is available, the system reduces this supply to zero and creates another supply record with a quantity of -10 and 01/01/2500 as the ship-by date. <p>Note: (Applicable for the Sterling Warehouse Management System) The ship-by date is not automatically calculated. It must be passed in the input.</p>
Default Expiration Days	<p>Enter the number of days that the item is available after it is received. After this period of time, the item is no longer considered as available inventory.</p> <p>For example, if you sell roses and know they expire 3 days after you have received them, enter 3.</p>
ATP Rule	<p>Select an ATP Rule to use for this item. The parameters defined in ATP rules are used to determine the available inventory for this item. For more information about ATP rules, see the <i>Sterling Global Inventory Visibility: Configuration Guide</i>.</p> <p>Note: If you do not choose an ATP rule, the system's DEFAULT ATP rule is used.</p>
ATP Monitor Rule	<p>Select an Inventory Monitoring Rule to use for this item. The minimum inventory levels for this item are based on the parameters set up for the selected ATP Monitoring Rule. For more information about ATP monitoring rules, see the <i>Sterling Global Inventory Visibility: Configuration Guide</i>.</p>

Table 4–6 Inventory Info Tab

Field	Description
Node Level ATP Monitor Rule	Select an Node Level Inventory Monitoring Rule to use for this item. The minimum inventory levels for this item are based on the parameters set up for the selected ATP Monitoring Rule. For more information about ATP monitoring rules, see the <i>Sterling Global Inventory Visibility: Configuration Guide</i> .
Default Product Class	Select any of the configured product classifications based on your business practices. For more information about product classes, see the <i>Selling and Fulfillment Foundation: Application Platform Configuration Guide</i> .
Track FIFO in Inventory	Indicates that inventory needs to be tracked using the First In First Out (FIFO) number. The FIFO number is automatically generated by the system.
Lead Days	Enter the amount of time (in days) needed to procure the item and make it available for shipping.
Future Safety Factor Percentage	Enter the percentage of inventory for this item that you want to exclude from future inventory availability.
Primary Supplier	Enter the primary supplier for this item.
Onhand Safety Factor Percentage	Select this option and enter the percentage of inventory for this item that you want to exclude from on hand inventory availability. If this option is chosen, you cannot specify an Onhand Safety Factor Quantity.
Onhand Safety Factor Quantity	Select this option and enter the quantity of inventory for this item that you want to exclude from on hand inventory availability. If this option is chosen, you cannot specify an Onhand Safety Factor Percentage.
Note: If any safety factor value (quantity or percentage) is available from the item's classification, but an on hand safety factor value (quantity or percentage) is specified at the item level, Selling and Fulfillment Foundation always uses the value specified at the item level, and does not inherit the value from the item's classification.	
Serial Tracked	Select Serial Tracked to enable location-level tracking by serial number in the Sterling WMS.

Table 4–6 Inventory Info Tab

Field	Description
Serialized	Select Serialized to ensure that serial numbers are captured by the system during the outbound and return processes.
Number of Secondary Serials	Enter the number (0-9) of secondary serial numbers you want to use for tracking by serial number.
Item Based Allocation Allowed	Select the appropriate option to allow item-based allocation for an item from the drop-down list. When the 'Use Item Based Allocation' rule is enabled, the item-based allocation is applicable only for the items and nodes that 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> .
Use Unplanned Inventory	Select the appropriate option to allow unplanned inventory for this item to be used. This configuration can also be inherited from item classification. For more information see Defining Classification Definitions .
Tag Controlled	
Not Tag Controlled	Select Not Tag Controlled if you do not want to use a tag number to uniquely identify this item in the system.
Sometimes Tag Controlled	Select Sometimes Tag Controlled if you want to be able to assign a tag number to identify an item in special circumstances, such as customer pickup.
Always Tag Controlled	Select Always Tag Controlled if you always want to use a tag number to uniquely identify different instances of this item in the system.
Tag Identifiers	
	This table displays when either Sometimes Tag Controlled or Always Tag Controlled are selected. In the table select one or more attributes to be used as tag identifiers for the item. The combination of the tag identifiers generates the tag number that the system uses to identify tag-controlled items.

Table 4–6 Inventory Info Tab

Field	Description
Tag Attributes	
	<p>This table displays when either Sometimes Tag Controlled or Always Tag Controlled are selected. In the table select one or more attributes to be used as tag descriptors for the item.</p> <p>The tag attributes do not generate a tag number. They are used for informational purposes, and only the attributes that have been checked display in the Application Console UI.</p>

4.1.2.5 Viewing Inheritable Attributes

Some attributes located on the Sourcing and Transportation Info, Inventory Info, and Extended Attributes windows, can be specified as common item attributes at the classification level. For more information about defining common item attributes, see [Section 4.6.3.4, "Defining Common Item Attributes for a Classification Value"](#).

To view inheritable attributes:

1. In the Item Details window, choose . The Inheritable Attributes pop-up window displays.

 Inheritable Attributes			
Name	Value	Classification Path	Is Inherited
Onhand Safety Factor Quantity	10.0	/Class1/CV1	N
ATP Monitor Rule	new	/Class1/CV1	N
ATP Rule	DEFAULT	/Class1/CV1	Y

Table 4–7 Inheritable Attributes Pop-up Window

Field	Description
Name	The name of the attribute.
Value	The value specified for the attribute at the classification level.

Table 4–7 Inheritable Attributes Pop-up Window

Field	Description
Classification Path	This path indicates the location of the classification where the common item attribute has been specified.
Is Inherited	Indicates whether the value for this attribute is inherited for this item.

4.1.2.6 Defining a Product Item’s Classifications

You can set up codes and classifications necessary for shipping an item. The fields within this screen can be entered as free form text or classification hierarchies can be created so that the values of the fields can be selected from a list of pre-defined values. For more information about classifications and classic at ion hierarchies, see [Section 4.6, "Defining Classifications"](#).

To set up an item’s classifications:

1. In the Item Details window, choose the Classifications tab.
2. Enter information in the applicable fields. Refer to [Table 4–8](#) for field value descriptions.

Table 4–8 Classifications Tab

Field	Description
Standard Classification	
Harmonized Code	Enter the item's harmonized code. If the classification purpose has been defined for the purpose definition 'Harmonized Code', the  icon displays. Click it to display the classification definition that it has been bound to.
Schedule B Code	Enter the item's schedule B code. If the classification purpose has been defined for the purpose definition 'Schedule B Code', the  icon displays. Click it to display the classification definition that it has been bound to.
Commodity Code	Enter the item's commodity code. If the classification purpose has been defined for the purpose definition 'Commodity Code', the  icon displays. Click it to display the classification definition that it has been bound to.
NMFC Code	Enter the item's National Motor Freight Code. If the classification purpose has been defined for the purpose definition 'NMFC Code', the  icon displays. Click it to display the classification definition that it has been bound to.
ECCN Number	Enter the item's Export Commodity Control Number. If the classification purpose has been defined for the purpose definition 'EECN Number', the  icon displays. Click it to display the classification definition that it has been bound to.
NAICS Code	Enter the item's North American Industry Classification System code. If the classification purpose has been defined for the purpose definition 'NAICS Code', the  icon displays. Click it to display the classification definition that it has been bound to.
UNSPSC	Enter the item's United Nations Standard Product and Services Code. If the classification purpose has been defined for the purpose definition 'UNSPC', the  icon displays. Click it to display the classification definition that it has been bound to.
NMFC Class	The National Motor Freight Code Class of the item.

Table 4–8 Classifications Tab

Field	Description
Tax Product Code	Enter the item's tax product code.
Cost Posting Classification	The Cost Posting Classification of the item.
Hazardous Material Class	Enter the item's hazardous material class. If the classification purpose has been defined for the purpose definition 'HazmatClass', the  icon displays. Click to display the classification definition that it has been bound to.
Is Hazardous Item	If the hazardous material class is a valid hazmat compliance entry, when you click  , this check box is automatically selected to indicate that the item is a hazardous item. Select the check box, if applicable.
Model	Enter the model number of the item.
Organizational Classification	
Product Line	Enter the product line as per your business practices. If the classification purpose has been defined for the purpose definition 'Product Line', the  icon displays. Click it to display the classification definition that it has been bound to.
Item Type	Enter the item type as per your business practices. If the classification purpose has been defined for the purpose definition 'Item Type', the  icon displays. Click it to display the classification definition that it has been bound to.
Operational Classification	
Storage Type	Enter the item's storage code. The storage code indicates any special attributes of the item that may require it to be stored at a certain location in a warehouse. If the classification purpose has been defined for the purpose definition 'Storage Type', the  icon displays. Click it to display the classification definition that it has been bound to.
Picking Type	This field is not used in this version.

Table 4–8 Classifications Tab

Field	Description
Velocity Code	Enter the item's velocity code. The velocity code indicates the item's turn over rate in a warehouse.
Operational Configuration Complete	This field is not used in this version.
Hazmat Information	<p>This panel displays when a value is entered in the Hazardous Material Class field and the Is Hazardous Item checkbox is selected.</p> <p>If the value entered in the Hazardous Material Class field is valid, then the Hazmat Information panel displays with values populated.</p> <p>If the value entered in Hazardous Material Class field is invalid, then the Hazmat Information panel displays with blank values.</p>
Symbols	Indicates the hazmat symbol of the hazardous item.
Proper Shipping Name	The proper shipping name or description of the hazardous item.
Hazard Class	The item's hazard class.
UN Number	This is a four-digit number, which identifies hazardous items (such as explosives and poisonous materials) of commercial importance. This numbering scheme is widely used in international commerce. For instance, to label the contents of shipping containers.
Packing Group	Indicates the degree of danger of a hazardous item within its hazard class or division, and the type of packaging required for the item. The valid values are: Packing Group I - high danger, Packing Group II - medium danger, Packing Group III - Low danger.
Label Code	The label code of the hazardous item.
Special Provisions	Special provisions to be provided for shipping the hazardous item.
Exception	The packing exception for shipping the hazardous item.

4.1.2.7 Defining a Product Item's Additional Attributes

You can set up any additional attributes created in Catalog Configurations. You can also configure any aliases and item exclusion codes.

To define an item's additional attributes:

1. In the Item Details window, choose the Other Attributes tab.
2. Enter information in the applicable fields. Refer to [Table 4–9](#) for field value descriptions.

The screenshot shows the 'Other Attributes' tab selected in the Item Details window. The window has a top navigation bar with tabs: Other Attributes (selected), Associations, Node Items, Available Services, and Item Instructions. Below this is a sub-navigation bar with tabs: Primary Info (selected), Units Of Measure, Inventory Info, and Classifications. The main content area is divided into three sections:

- Additional Attributes:** Contains two input fields: 'Attr2' and 'Reference'.
- Aliases:** Contains one input field: 'UPC Code'.
- Item Exclusion Codes:** Contains two lists, 'Available' and 'Subscribed', each with an 'Item Exclusion Code' header. The 'Available' list contains the entry 'DEFAULT'. Between the two lists are right and left arrow buttons.

Table 4–9 Other Attributes Tab

Field	Description
Additional Attributes	Any additional attributes that have been created in Catalog Configurations. Enter information as per your business practices. For more information about creating additional attributes for an item, see Section 3.2, "Defining Additional Attributes" .
Aliases	Any aliases that have been created in Item Attributes. An item's alias is an identifier for an item. Enter information as per your business practices. For more information about creating aliases, see Section 4.7, "Defining Item Attributes" .
Item Exclusion Codes	<p>Any item exclusion codes that have been created in Item Attributes. An item exclusion code may be set up if your business practices involve the regulation of item shipments for certain countries. For example, if there are certain countries that you ship to that do not allow items with certain hazardous materials to be shipped to them, you may create an item exclusion code that you can associate with any of your items that may contain the certain hazardous material.</p> <p>You can add and remove pertinent item exclusion codes using the arrow buttons.</p> <p>For more information about creating item exclusion codes, see Section 4.7, "Defining Item Attributes".</p>

The screenshot shows a pop-up window titled 'Item Instruction Details'. It contains the following elements:

- Item Instruction Type:** A dropdown menu with a gear icon to its right.
- Sequence No:** A text input field.
- Use Item Instruction Code:** A radio button that is selected.
- Item Instruction Code:** A dropdown menu with a plus icon to its right.
- Create New Item Instruction:** A radio button that is unselected.
- Text:** A large text area for entering the instruction text.

Table 4–10 *Item Instruction Details Pop-Up Window*

Field	Description
Item Instruction Type	Select the item instruction type.
Sequence No	If there are multiple, sequential instructions for the product item, enter the sequence of the instruction you are adding.
Use Item Instruction Code	Choose Use Item Instruction Code if you want to use a pre-existing item instruction code.
Item Instruction Code	If you chose Use Item Instruction Code, select the applicable item instruction code.
Create New Item Instruction	Select Create New Item Instruction if no pre-existing item instruction code exists and you want to create a free-form text instruction.
Text	If you selected Create New Item Instruction, enter the instruction.

4.1.2.7.1 Modifying a Product Item Instruction

To modify a product item instruction:

1. In the Product item Details window, choose the Item Instructions tab.
2. From the Item Instructions table, select the applicable item instruction choose . The Item Instruction Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 4–14](#) for field value descriptions.
4. Choose .

4.1.2.7.2 Deleting a Product Item Instruction

To delete a product item instruction:

1. In the Product Item Details window, choose the Item Instructions tab.
2. From the Item Instructions table, select the applicable item instruction choose .

4.1.2.8 Defining Product Item Associations

You can set up cross-sell, up-sell, supersession, alternative, and competitive associations that apply to the item. You can set up an item's association on an item-by-item basis or by queries under which particular items fall.

Note: Bundle parents cannot have product item associations.

Cross-Sell

A cross-sell promotes an add-on or accessory item that, when combined with the primary item, makes a “better” package. For example, there is a shirt in your inventory that matches a pair of shoes. When you create the shoes item in the catalog, you associate the shirt as a cross-sell. This way when the customer orders the shoes it is possible to notify them about the matching shirt.

Up-Sell

An up-sell promotes a more powerful or higher-level item. For example, in your inventory you have a certain model of running shoe that comes in one of two ways, an inexpensive, lower-quality version or an expensive, top-of-the-line version. When you create the catalog item for the lower-quality shoes you associate the more expensive shoes as an up-sell. When the customer goes to order the lower-quality shoes it is possible to notify them about the higher-quality shoes.

Supersession

Setting up a supersession association provides the ability to recognize that an ordered item can be substituted with one or more other items. An item is superseded when its effective date becomes obsolete. Typically, obsolete items are not offered to customers who perform a regular search of your business's Web site. However, if a customer performs an advanced search, looking specifically for obsolete items, he is offered the next item in the chain of supersession. Additionally, if an item becomes unavailable, such as an item that has sold out, the system tries to order the next item in the chain of supersession. For example, in your inventory you have a very popular brand of shoe that sells out quickly. There are other comparable brands of shoes that are popular but don't sell as much. When you create a catalog item for the popular brand of shoes you associate the other shoes as supersessions. When a customer orders the popular brand of shoes that are not in stock, they can be given the option to substitute for one of the other shoes.

Preferred Substitutes

A preferred substitute is an item that is fulfilled prior to fulfilling the original item. The system checks and fulfills the order by shipping the preferred substitute item even when the original item is available.

Alternative

An alternative association provides the ability to choose an item that is very similar to the original as an alternate. For example, in your inventory, you have a popular model of running shoe in two different models, high-top or low-top. When a customer chooses the high-top sneaker, they are given a choice to choose the low-top as an alternative.

Competitive

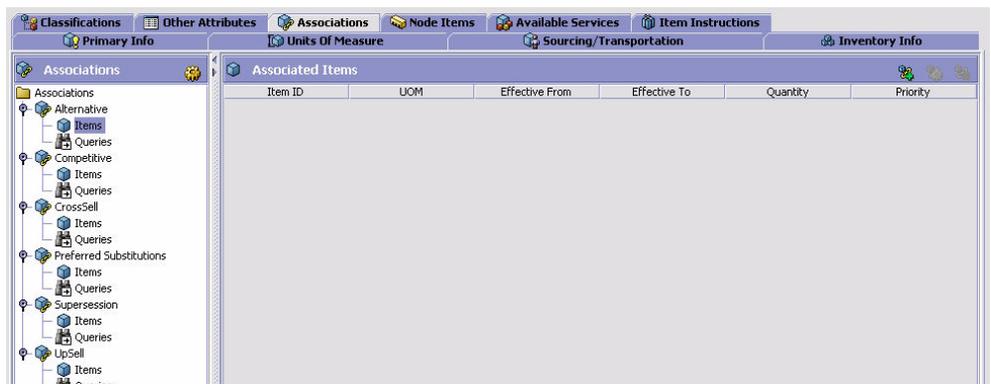
A competitive association provides the ability to associate similar items from different manufacturers. You can also use this association to track products from competitors. For example, a competitive item to a Company X high-top basketball shoe is a Company Y high-top basketball shoe.

You can set up an item's association in two different ways:

- [Association by Item](#)
- [Association by Query](#)

4.1.2.8.1 Association by Item

1. In the Item Details window, click on the Associations tab.



2. Expand the branch of the association type you want to set up for the item. You can also double-click on items or queries to display the appropriate list.
3. Highlight Items and click on . Refer to [Table 4–11](#) for field value descriptions. Enter information in the applicable fields.
4. Click on .

The screenshot shows a dialog box titled "Associated Item". It has the following fields:

- Item ID: A dropdown menu with a search icon on the right.
- UOM: A dropdown menu.
- Effective Date From: A date picker.
- Effective Date To: A date picker.
- Quantity: A text input field.
- Priority: A text input field.

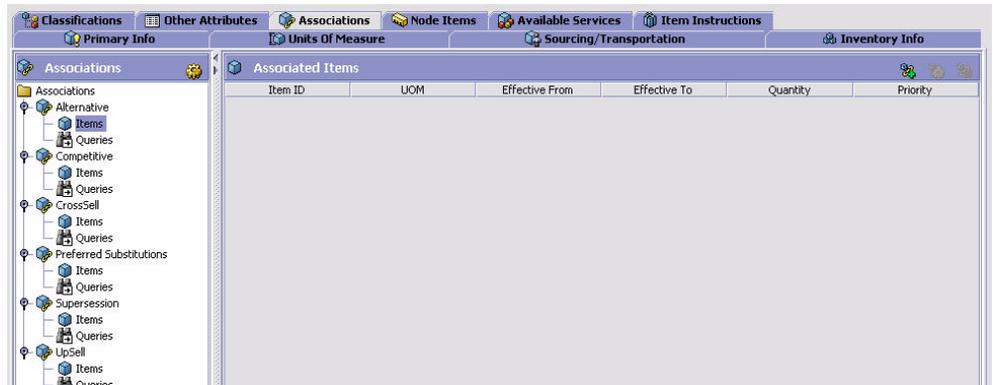
At the bottom right, there are two buttons: "OK" and "Cancel".

Table 4–11 Association by Items

Field	Description
Item ID	Enter the item's ID.
UOM	Select the item's UOM from the drop-down menu.
Effective Date From	Enter the date from which the association is valid.
Effective Date To	Enter the date until which the association is valid.
Quantity	Enter the quantity of that item required for the association to be valid. For example, one could say that two 1-litter bottles of water are a supersession for one 2-litter bottle of water.
Priority	Enter the priority the item has over another associated item.

4.1.2.8.2 Association by Query

1. In the Item Details window, click on the Associations tab.



2. Expand the branch of the association type you want to set up for the item.
3. Highlight Query and click on . Enter information in the applicable fields. Refer to [Table 4–12](#) for field value descriptions.
4. Click on .

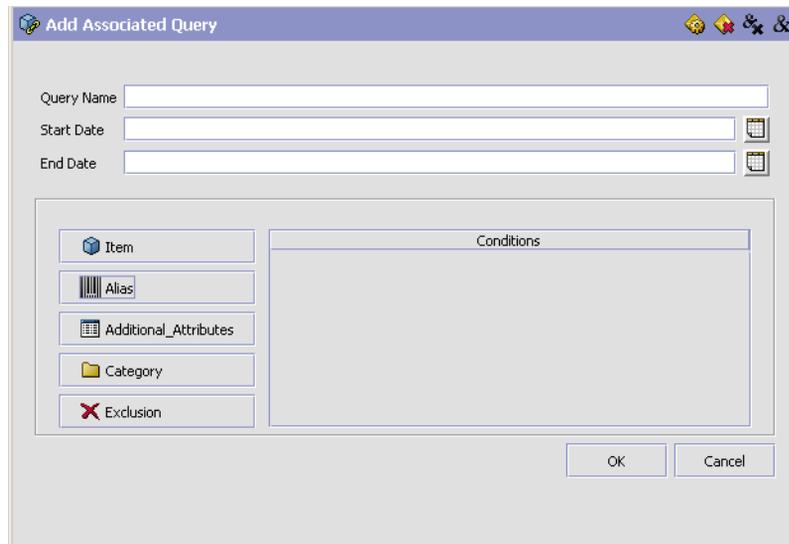


Table 4–12 Association by Queries

Field	Description
Query Name	Enter the name of the query.
Start Date	Enter the date from which the association is valid.
End Date	Enter the date until which the association is valid.
Item Query Window	Use this window to create the item query you want to create the association with. For more information about creating item queries, see Section 4.1.5, "Using the Condition Builder" .

4.1.2.9 Defining a Product Item’s Node Item Attribute Overrides

You can define item attribute overrides at a given node within your organization. The attributes you assign at the node level override any of the associated attributes that have been defined for an item when an order containing that item is released to the node. For more information about defining item attribute overrides at the node level, see [Section 4.2, "Defining Item Attribute Overrides at the Node Level"](#).

4.1.2.10 Defining a Product Item’s Associated Service Items

You can associate service items that can optionally be ordered when this product item is ordered. For example, you can create a Dishwasher product item and associate a 2-man Delivery delivery service item and an Installation provided service item with it. These service items are then available to the customer as additional options to them when they order the product item.

You can also associate other valued added services such as kitting service, which defines a process for how to assemble the item.

You can use the Available Services tab for:

- [Associating a Service Item with a Product Item](#)
- [Modifying Service Items Associated with a Product Item](#)
- [Deleting a Service Item Associated with a Product Item](#)

4.1.2.10.1 Associating a Service Item with a Product Item

To associate service items to a product item:

1. In the Item Details window, choose the Available Services tab.
2. In the Available Services tab, choose the Provided Services tab to associate a provided service item or the Delivery Services tab to associate a delivery service.
3. From the Available Services table, choose . The Service Association pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 4–13](#) for field value descriptions.
5. Choose .



Service Association

Item ID

UOM

Service Time Offset

Before Product Reaches Destination After Product Reaches Destination

Service Time Offset (Hours)

Hold Scheduling Until Associated Line Completes

Capacity

Service Item Capacity Quantity Per Product Quantity

When Pricing Quantity Is Derived Based On Quantity Of Associated Lines

Service Item Pricing Quantity Per Product Quantity

Service Skills For This Service Association

Service Skill ID	Description

Results 0 Of 0

Table 4–13 Service Association Pop-Up Window

Field	Description
Item ID	Select the service item ID.
UOM	Select the service item's unit of measure.
Service Time Offset	Enter the time difference between provided service and product delivery (in hours). For example, if you are configuring a Door product item that requires an Measurement provided service one day before it can be delivered using a 1-Man Delivery delivery service, you would set Service Time Offset to 24 hours.
Hold Scheduling Until Associated Line Completes	Select Hold Scheduling Until Associated Line Completes if you do not want the service item to be scheduled until the product item has been shipped.
Capacity	
Service Item Capacity Quantity Per	Enter the resource pool capacity that is used up for a defined ordered product quantity as per the service items unit of measure. For example, if it takes one hour to install one dishwasher, you can configure an Installation provided service with a unit of measure of hours to consume one hour of resource capacity per one Dishwasher product item ordered.
Product Quantity	Enter the quantity of the product item against which capacity quantity is consumed.
When Pricing Quantity Is Derived Based On Quantity Of Associated Lines	
Service Item Pricing Quantity Per	Enter how much of the service pricing unit of measure is used per the specified product quantity. For example, you are defining an Installation provided service for a Carpet product item and the pricing unit of measure for Installation is HOURS. It takes 2 HOURS per 300 SQUARE FEET of Carpet. You would enter '2' in this field.
Product Quantity	Enter the quantity of the product item against which pricing quantity is determined.

Table 4–13 Service Association Pop-Up Window

Field	Description
Service Skills for This Service Association	
Service Skill ID	List of service skill IDs for the item service association. For information about adding service skills to an item service association, see Section 4.1.2.10.4, "Adding a Service Skill to a Service Association" . For information about removing service skills from an item service associations, see Section 4.1.2.10.5, "Removing a Service Skill from a Service Association" .
Description	Brief description of each service skill listed.

4.1.2.10.2 Modifying Service Items Associated with a Product Item

To modify service items associated with a product item:

1. In the Item Details window, choose the Available Services tab.
2. In the Available Services tab, choose the Provided Services tab to modify associated provided service items or the Delivery Services tab to modify associated delivery services.
3. From the Available Services table, select the applicable service item association and choose . The Service Association pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 4–13](#) for field value descriptions.
5. Choose .

4.1.2.10.3 Deleting a Service Item Associated with a Product Item

To delete a service item associated with a product:

1. In the Item Details window, choose the Available Services tab.
2. In the Available Services tab, choose the Provided Services tab to delete associated provided service items or the Delivery Services tab to delete associated delivery services.
3. From the Available Services table, select the applicable service item association and choose .

4.1.2.10.4 Adding a Service Skill to a Service Association

You can add multiple service skills to a service association. The service skill table for the item service association manages service skills for that service association.

To add a service skill to a service association:

1. In the Service Association Details window, choose  from the "Service Skills For This Service Association" table. The Service Skills List pop-up window displays.
2. Select the applicable service skills and choose . The service skill is added to the Service Skills table.

4.1.2.10.5 Removing a Service Skill from a Service Association

To remove a service skill from a service association, in the Service Association Details window, select the applicable service skill from the "Service Skills For This Service Association" table and choose .

4.1.2.11 Defining a Product Item's Instructions

You can define any special instructions that pertain to a product item. For example, you can associate a special instruction of Handle With Care with a fragile product item.

You can use the Item Instructions tab for:

- [Adding a Product Item Instruction](#)
- [Modifying a Product Item Instruction](#)
- [Deleting a Product Item Instruction](#)

4.1.2.11.1 Adding a Product Item Instruction

To add a product item instruction:

1. In the Item Details window, choose the Item Instructions tab.
2. From the Item Instructions table, choose . The Item Instruction Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 4–14](#) for field value descriptions.
4. Choose .

Table 4–14 *Item Instruction Details Pop-Up Window*

Field	Description
Item Instruction Type	Select the item instruction type. For example, a Handling item type indicates the instruction pertains to handling the product item.
Sequence No	If there are multiple, sequential instructions for the product item, enter the sequence of the instruction you are adding.
Use Item Instruction Code	Choose Use Item Instruction Code if you want to use a pre-existing item instruction code. For example, you could have a pre-existing Handle With Care item instruction code that you can associate with fragile items.
Item Instruction Code	If you chose Use Item Instruction Code, select the applicable item instruction code.

Table 4–14 Item Instruction Details Pop-Up Window

Field	Description
Create New Item Instruction	Select Create New Item Instruction if no pre-existing item instruction code exists and you want to create a free-form text instruction.
Text	If you selected Create New Item Instruction, enter the instruction.

4.1.2.11.2 Modifying a Product Item Instruction

To modify a product item instruction:

1. In the Item Details window, choose the Item Instructions tab.
2. From the Item Instructions table, select the applicable item instruction choose . The Item Instruction Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 4–14](#) for field value descriptions.
4. Choose .

4.1.2.11.3 Deleting a Product Item Instruction

To delete a product item instruction:

1. In the Item Details window, choose the Item Instructions tab.
2. From the Item Instructions table, select the applicable item instruction choose .

4.1.2.12 Defining a Product Item's Children Items

If an item is marked as an item with variations, you can define its child items. Child items are items which share common attributes with the item with variations. For example, if the item with variations is a Short-sleeved Polo shirt, possible child items are Small Red Short-sleeved Polo Shirt and Medium Red Short-sleeved Polo Shirt.

You can use the Children Items tab for:

- [Adding a Child Item](#)
- [Removing a Child Item](#)

4.1.2.12.1 Adding a Child Item

To add a child item:

1. In the Item Details window, choose the Primary Info tab.
2. If it is not already checked, check the Is Item With Variations box.
3. Choose the Children Items tab.
4. From the Children Items table, choose . The Product Item Search displays.
5. Search for the desired item and choose .
6. Click Save, to save the Item With Variations.

4.1.2.12.2 Removing a Child Item

To remove a child item:

1. In the Item Details window, choose the Children Items tab.
2. From the Children Items table, select the item you wish to remove.
3. Choose .

4.1.2.13 Defining a Product Item's Components

If you have identified an item as a kit item by associating a kit code to it from the Primary Info tab, the Components tab can be used from the Item Details window. On the Components tab you can create a list of items that make up the kit.

Important: In Selling and Fulfillment Foundation, orders only support one-tier level for all types of kits except bundles. The Applications Manager, however, does not prevent you from configuring other kits with multiple-tier levels.

You can use the Components tab for:

- [Adding a Component to an Item](#)
- [Modifying an Item's Component](#)
- [Deleting an Item's Component](#)

4.1.2.13.1 Adding a Component to an Item

To add a kit component to an item:

1. In the Item Details screen, choose the Components tab.
2. If you are using the Selling and Fulfillment Foundation inventory costing feature and you have identified the kit as a physical kit, select the cost factor group you want applied for work-in-progress cost computations from the Cost Factor Group to be Used for the Physical Kit Cost Calculations drop-down list. This list contains the cost factor groups defined by the catalog organization's Primary Enterprise.
3. If you identified the kit as a bundle, select the fulfillment mode for the bundle. Choose one of the following:
 - Ships Together - The components in this bundle can be procured from different nodes and then merged at the shipping node to be shipped as a single shipment.
 - Ships Independently - the components in this bundle are treated as independent lines, shipping when they are available. As such, they may be delivered on different days.
 - Delivers Together - this bundle can be shipped from multiple nodes. Delivery schedules are driven by the expected delivery date. Bundle components are shipped according the latest expected delivery date of a component, so that all components can be delivered together.

4. From the Components table, click .
5. The Component Details pop-up window is displayed.
6. Enter information in the applicable fields. Refer to [Table 4–15](#) for field value descriptions.
7. Choose .

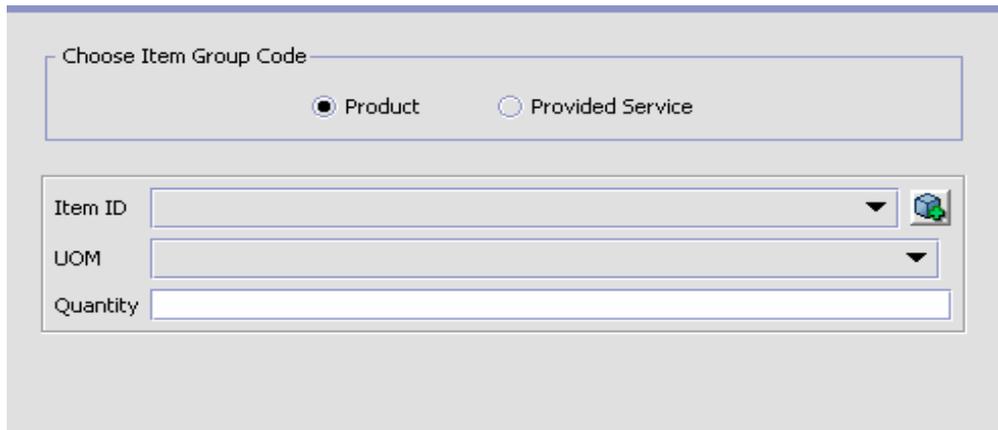
Table 4–15 Component Details Pop-Up Window - Physical and Logical Kits

Field	Description
Item ID	Select the item you want to add to the kit.
UOM	Select the unit of measure of the item you are adding to the kit.
Quantity	Enter the quantity of the item you are adding to the kit.

4.1.2.13.2 Modifying an Item’s Component

To modify an item’s kit component:

1. In the Item Details window, choose the Components tab.
2. From the Components table, select the applicable kit component and choose . The Component Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 4–16](#) for field value descriptions.
4. Choose .



Choose Item Group Code

Product Provided Service

Item ID 

UOM

Quantity

Table 4–16 Component Details Pop-Up Window - Bundle

Field	Description								
Product	Choose this option to input details about the product. <table border="1" data-bbox="686 371 1279 661"> <thead> <tr> <th>Field</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>Item ID</td> <td>Select the item you want to add to the kit.</td> </tr> <tr> <td>UOM</td> <td>Select the unit of measure of the item you are adding to the kit.</td> </tr> <tr> <td>Quantity</td> <td>Enter the quantity of the item you are adding to the kit.</td> </tr> </tbody> </table>	Field	Description	Item ID	Select the item you want to add to the kit.	UOM	Select the unit of measure of the item you are adding to the kit.	Quantity	Enter the quantity of the item you are adding to the kit.
Field	Description								
Item ID	Select the item you want to add to the kit.								
UOM	Select the unit of measure of the item you are adding to the kit.								
Quantity	Enter the quantity of the item you are adding to the kit.								
Provided Service	Choose this option to input details regarding provided service. <table border="1" data-bbox="686 793 1279 1083"> <thead> <tr> <th>Field</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>Item ID</td> <td>Select the item you want to add to the kit.</td> </tr> <tr> <td>UOM</td> <td>Select the unit of measure of the item you are adding to the kit.</td> </tr> <tr> <td>Quantity</td> <td>Enter the quantity of the item you are adding to the kit.</td> </tr> </tbody> </table>	Field	Description	Item ID	Select the item you want to add to the kit.	UOM	Select the unit of measure of the item you are adding to the kit.	Quantity	Enter the quantity of the item you are adding to the kit.
Field	Description								
Item ID	Select the item you want to add to the kit.								
UOM	Select the unit of measure of the item you are adding to the kit.								
Quantity	Enter the quantity of the item you are adding to the kit.								

4.1.2.13.3 Deleting an Item’s Component

To delete an item’s kit component:

1. In the Item Details window, choose the Components tab.
2. From the Components table, select the applicable kit component and choose .

4.1.2.14 Defining a Product Item’s Container Attributes

An item can be a container item, which is used to ship other items. For more information about creating an item that is a container, see

Section 4.1.1, "Creating a Product Item". You can specify the container attributes for an item.

You can use the Container Attributes tab for:

- [Adding a SKU Capacity Definition](#)
- [Modifying a SKU Capacity Definition](#)
- [Deleting a SKU Capacity Definition](#)

To define an item's container attributes:

1. In the Item Details window, choose the Container Attributes tab.
2. Enter information in the applicable fields. Refer to [Table 4-17](#) for field value descriptions.

The screenshot shows the 'Container Attributes' tab in a software application. The interface includes several sections for defining container properties:

- Container Restrictions:** Fields for 'Reserved for the following Buyers' and 'Reserved for Items having the following Containerization Categories', each with a search icon and a text input field.
- Container Volume:** Fields for 'Capacity Volume' and 'Volume Allowance', each with a text input field and a dropdown menu.
- Maximum weight that can be placed in container:** A text input field with a dropdown menu.
- Capacity of the container is defined in terms of volumes except for the following products:** A section for defining exceptions.
- Container SKU Capacity Definition:** A table with two columns: 'SKU Unit Size Code' and 'Quantity'. The table is currently empty.

SKU Unit Size Code	Quantity
--------------------	----------

Table 4–17 Container Attributes Tab

Field	Description
Container Restrictions	
Reserved for the following Buyers	Specify the buyers to whom the container should be reserved. This prevents the use of these containers by other buyers. Note: Buyers with containers reserved for their use can only use the reserved containers. They cannot make use of other containers from the common pool.
Reserved for Items having the following Containerization Categories	Select Containerization Categories if you want to reserve the use of this container to items that have been categorized in specific ways. For example, you can restrict the use of this container to items that are categorized as "non-perishable" for the containerization category. See Section 4.6, "Defining Classifications" .
Container Volume	
Capacity Volume	Enter the cubic amount that can be placed in this container, and select the cubic unit of measure for that volume (for example, cubic foot, liter).
Volume Allowance	Enter the cubic amount of additional items that can be added when there is are a few remaining items to be containerized. For example, a container has a capacity volume of 100 cubic feet. Two containers have been filled, and there are a few remaining items to be containerized. The remaining items have a volume of two cubic feet. If the Volume Allowance is set to accept two cubic feet, or more, the remaining items are placed in the last container. Leave this field blank to specify no Volume Allowance.
Maximum weight that can be placed in the container	Enter the maximum weight permitted for this container, and select the unit of measure used for that weight.
Container SKU Capacity Definition	

Table 4–17 Container Attributes Tab

Field	Description
SKU Unit Size Code	Identifies SKU Unit Size Code used to identify items.
Quantity	How many items of the specified SKU Unit Size Code can be placed in this container. In some cases, it is easier to specify the number of items that can be placed in the container than it is to attempt to calculate the number of items based on the size of the items and the volume of the container. For example, you might want to specify the capacity of the container for irregularly shaped items, such as balls, or highly compressible items, such as shirts.

4.1.2.14.1 Adding a SKU Capacity Definition

A SKU capacity definition allows you to configure how many items with a given classification can be placed in this container. The classification is established by binding the "SKU Unit Size Code" to a classification definition. See [Section 4.6.1, "Defining Classification Definitions"](#).

To add a SKU Capacity Definition:

1. In the Item Details window, choose the Container Attributes tab.
2. From the Container SKU Capacity Definition table, choose . The SKU Capacity Definition pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 4–18](#) for field value descriptions.
4. Choose .

The screenshot shows a pop-up window with a light gray background. It contains two input fields. The first field is labeled 'Harmonized Code' and has a small icon of a person to its right. The second field is labeled 'Quantity'.

Table 4–18 *SKU Capacity Code Details Pop-up Window*

Field	Description
Classification Type	<p>A field used in Classifications is mapped to the SKU unit size code as part of the configuration for Items. The field name displayed here, "Harmonized Code", is an example. See Section 4.6, "Defining Classifications".</p> <p>Enter a value used in classification to select items that have that classification.</p>
Quantity	Specify how many of this item classification can be placed in the container.

4.1.2.14.2 Modifying a SKU Capacity Definition

To modify a SKU Capacity Definition:

1. In the Item Details window, choose the Container Attributes tab.
2. From the Container SKU Capacity Definition table, select the applicable SKU capacity definition and choose . The SKU Capacity Code Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 4–18](#) for field value descriptions.
4. Choose .

4.1.2.14.3 Deleting a SKU Capacity Definition

To delete a SKU Capacity Definition:

1. In the Item Details window, choose the Container Attributes tab.
2. From the Container SKU Capacity Definition table, select the applicable SKU capacity definition and choose .

4.1.2.15 Defining a Product Item's Extended Attributes

Selling and Fulfillment Foundation allows you to create your own extended item attributes. For more information about creating extended item attributes, see the *Selling and Fulfillment Foundation: Customization Guide*.

These attributes can also be specified at the classification level. For more information about viewing inheritable attributes, see [Section 4.1.2.5, "Viewing Inheritable Attributes"](#).

For more information about defining common item attributes, see [Section 4.6.3.4, "Defining Common Item Attributes for a Classification Value"](#).

To define a product item's extended attributes:

1. In the Item Details window, choose . The Extended Attributes pop-up window displays.
2. Enter information into the applicable fields.
3. Click OK. This closes the Extended Attributes pop-up window and returns you to the Item Details window.
4. Choose .

4.1.3 Deleting an Item

Important: Before you delete an item you must consider the fact that the item may be used in existing transaction data for inventory, orders, ship advices, returns, distribution rules, and/or price sets. Also, depending on your configuration, external systems may also be referencing this item.

To delete an item:

1. From the tree in the application rules side panel, choose Products > Items. The Product Item Search window displays in the work area.
2. Enter applicable search criteria and choose . The Item List displays.
3. Select the applicable item and choose .

4.1.4 Recomputing an Item's Computed Unit Cost

To recompute the unit cost of an item:

1. From the menu bar, choose Applications > Catalog Management > Items.
The Item Search window displays.
2. Enter applicable search criteria and choose . The Item List displays.
3. To recompute unit cost for all items, choose .
4. To recompute unit cost for a specific item, select the applicable item, right-click, and choose .

4.1.5 Using the Condition Builder

You can create an item query that you can use to search for items and establish item associations. For more information about item associations, see [Section 4.1.2.8, "Defining Product Item Associations"](#).

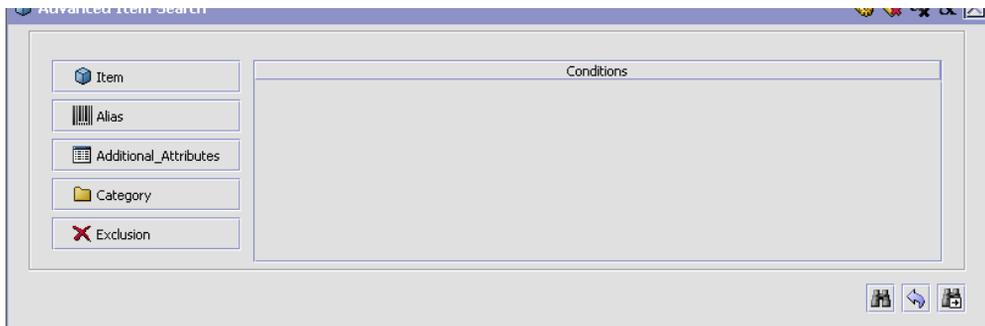
Using the condition builder, you can filter an item search by different query entities: item, item alias, additional item attributes, categories, and exclusion codes. Through each of these entities, you can use filters by attributes. For example, you can search for an item by its height measurement. You can also use action buttons to create priority and

logical operators. For example, to search for items by height and item ID but not by item type, you can group the Height and Item ID fields within parentheses.

4.1.5.1 Using the Condition Builder

To use the condition builder:

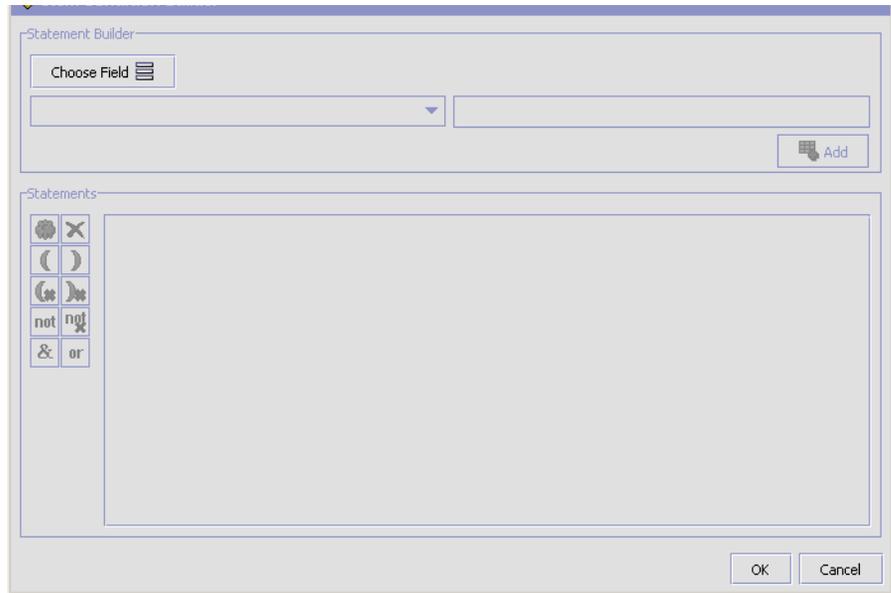
1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the applications side panel.
2. From the Catalog Management tree, choose Products > Items. The Product Item Search window displays in the work area.
3. Click . The Advanced Item Search window displays.



To create an item query:

4. Select a query entity

Click the query entity button for which you want to build a query. The condition builder popup window displays.

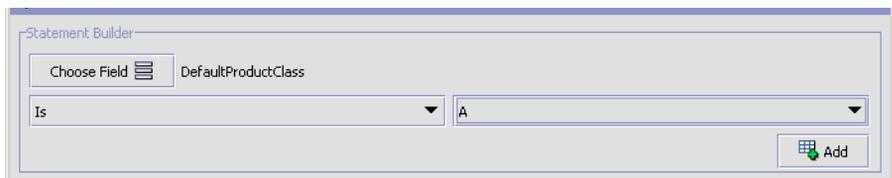


5. Select a statement

In the condition builder popup window, click on Choose Field and select the exact attribute you want as part of the query. For example, ItemID.

6. Select a query type and attribute value

In the drop down menu on the left, select the desired query type. In the field on the right, either select the desired attribute value from the drop down menu, or type the value if it is a text field. Click the Add button.



Repeat steps 5 and 6 until you have all of the statements you want in your condition.

By default, the statements are connected together through an AND operator.

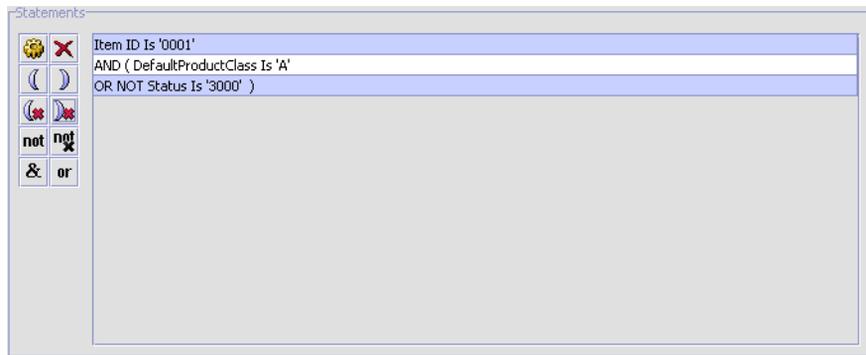
7. Optionally, edit statements by applying actions

If applicable, highlight the statement you want to edit and select the desired action. [Table 4–19, "Statement Actions"](#), describes the possible actions.

Table 4–19 Statement Actions

Action	Description
	Replace the highlighted statement. The Add button is replaced by a Replace button, and the statement can be selected again by going through Step 5.
	Delete the highlighted statement.
	Create a priority operator. For example, (A AND B) OR C as opposed to A AND (B OR C): The statements within the parenthesis are evaluated first.
	Remove a priority operator.
not	Create a negation operator. For example, A AND NOT B.
not	Remove a negation operator. For example, A AND NOT B would become A AND B.
&	Replace an OR operator by an AND operator.
or	Replace an AND operator by an OR operator.

Once the condition is built as desired, click OK.



8. Optionally, edit conditions by applying actions.

Repeat steps 4 through 7 until you have all the conditions you need for the query.

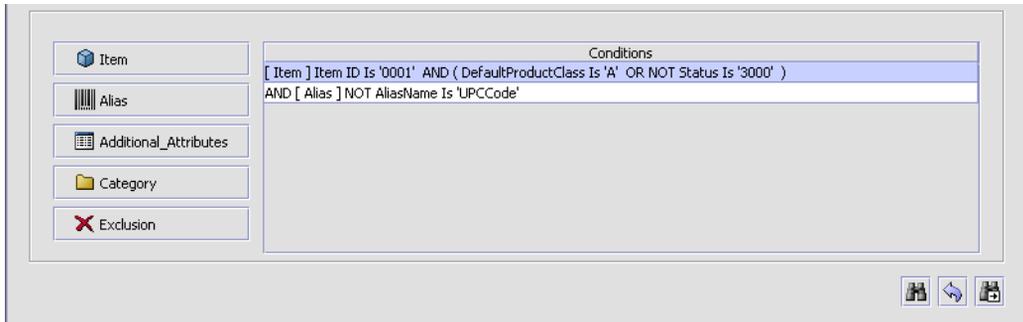
By default, the conditions are connected together through an AND operator.

If applicable, highlight the condition you want to edit and select the desired action located in the upper right-hand corner of the window. [Table 4–20, "Condition Actions"](#) describes the possible actions.

Table 4–20 Condition Actions

Action	Description
	Return to the condition builder popup window for that condition.
	Delete the highlighted condition.
	Replace an AND operator by an AND NOT operator.
	Replace an AND NOT operator by an AND operator.

Once the condition is built, click 



Important: When building queries using the Alias, Additional Attribute, and Exclusion entities, it is important to build conditions in which all attributes for that particular entity go into the same condition and not into multiple conditions.

4.1.5.2 Condition Builder Attributes

Each query entity allows you to build statements based on attributes defined throughout the Applications Manager.

- For the Item Condition Builder attributes, refer to [Table 4–21](#).
- For the Alias Condition Builder attributes, refer to [Table 4–22](#).
- For the Additional Attributes Condition Builder attributes, refer to [Table 4–23](#).
- For the Category Condition Builder attributes, refer to [Table 4–24](#).
- For the Exclusion Condition Builder attributes, refer to [Table 4–25](#).

Table 4–21 *Item Condition Builder Attributes*

Attribute	Description
Primary Info	
Kit Code	The kit code of the item, if applicable. Possible values are Physical Kit, Logical Kit, Bundle, and Dynamic Physical Kit.

Table 4–21 Item Condition Builder Attributes

Attribute	Description
Item ID	The ID of the item.
Default Product Class	The product class of the item.
Item UOM Master	The unit of measure of the item.
Status	The status of the item. Possible values are Held and Published.
Used As Container For Packing	The flag indicating whether or not the item is a container used for packing.
Global Item ID	The item's global identification number.
Item Type	The item's type.
Can Be Added To Work Order As Service Tool	The flag indicating whether or not the item can be added to a work order as a service tool.
Description	The detailed description of the item.
Short Description	The brief description of the item.
Product Line	The product line of the item.
Manufacturer Name	The name of the manufacturer who made the item.
Manufacturer Item	The identifier that the manufacturer uses to identify the item.
Manufacturer Item Description	The description that the manufacturer uses for the item.
Country of Origin	The country where the item was made.
Taxable Flag	The flag indicating whether or not the item is taxable.
Tax Product Code	The tax product code for this item.
Master Catalog ID	The identification number of the master catalog where the item is listed.
Minimum Order Quantity	The minimum quantity that an order can be placed for against the item.
Maximum Order Quantity	The maximum quantity that an order can be placed for against the item.
Measurements	
Unit Length	The length of the item.

Table 4–21 Item Condition Builder Attributes

Attribute	Description
Unit Length UOM	The unit of measure for the length of the item.
Unit Weight	The weight of the item.
Unit Weight UOM	The unit of measure for the weight of the item.
Unit Height	The height of the item.
Unit Height UOM	The unit of measure for the height of the item.
Unit Width	The width of the item.
Unit Width UOM	The unit of measure for the width of the item.
Inventory Parameters	
Lead Time	The amount of time (in days) needed to procure the item and make it available for shipping.
Tag Control Flag	The flag indicating to what extent the item is tag controlled. Possible values are Yes (always), No (never), and Sometimes.
Time Sensitive	The flag indicating whether or not the item is perishable with a specific ship-by date.
Don't Perform Inventory Check on Schedule and release.	The flag indicating whether or not an inventory check should be performed when scheduling or releasing this item. If this is set to 'Y', no inventory check occurs on schedule or release for this item. If this is set to 'N', inventory checks occur on schedule and release.
ATP Rule	The ATP rule that is used for this item. For more information about ATP rules, see the <i>Sterling Global Inventory Visibility: Configuration Guide</i> .
Inventory Monitor Rule	The inventory monitor rule used to monitor this item's inventory level.
Default Expiration Days	The number of days that an item is available after it is received.
Sourcing and Transportation Info	
Parcel Shipping Allowed	The flag indicating whether parcel shipping is allowed for transfer of items.
Requires Freezer	The flag indicating whether items require freezer storage during transportation.

Table 4–21 Item Condition Builder Attributes

Attribute	Description
Shipping By Air Allowed	The flag indicating whether items can be shipped by air.
Classification Codes	
Harmonized Code	The Harmonized Code of the item.
Schedule B Code	The Schedule B Code of the item.
ECCN No	The Export Commodity Control Number of the item.
NMFC Code	The National Motor Freight Code of the item.
NMFC Class	The National Motor Freight Code Class of the item.
UNSPSC	The United Nations Standard Product and Services Code of the item.
Hazmat Class	The Hazardous Material Class of the item.
Commodity Code	The commodity code of the item.

Table 4–22 Alias Condition Builder Attributes

Attribute	Description
Alias	
Alias Name	The name of the item's alias.
Alias Value	The value of the item's alias.

Table 4–23 Additional Attributes Condition Builder Attributes

Attribute	Description
Additional Attributes	
Name	The name of the item's additional attribute.
Value	The value of the item's additional attribute.

Table 4–24 Category Condition Builder Attributes

Attribute	Description
Category	
Category ID	The identification number of the item's category.
Organization Code	The organization that owns the item.
Category Path	The flattened path of the category tree for an item.

Table 4–25 Exclusion Condition Builder Attributes

Attribute	Description
Exclusion	
Exclusion Code	The item's exclusion code.

4.2 Defining Item Attribute Overrides at the Node Level

You can define item attribute overrides at a given node within your organization. The attributes you assign at the node level override any of the associated attributes that have been defined for an item when an order containing that item is released to the node.

You can use the Node Item Attributes branch for:

- [Creating Item Attribute Overrides at the Node Level](#)
- [Modifying Item Attribute Overrides at the Node Level](#)
- [Deleting Item Attribute Overrides at the Node Level](#)

4.2.1 Creating Item Attribute Overrides at the Node Level

To create item attribute overrides at the node level:

1. From the tree in the application rules side panel, choose Products > Node Item Attributes. The Node Items Search window displays in the work area.
2. Choose . The Create Item Node Definition pop-up window displays.

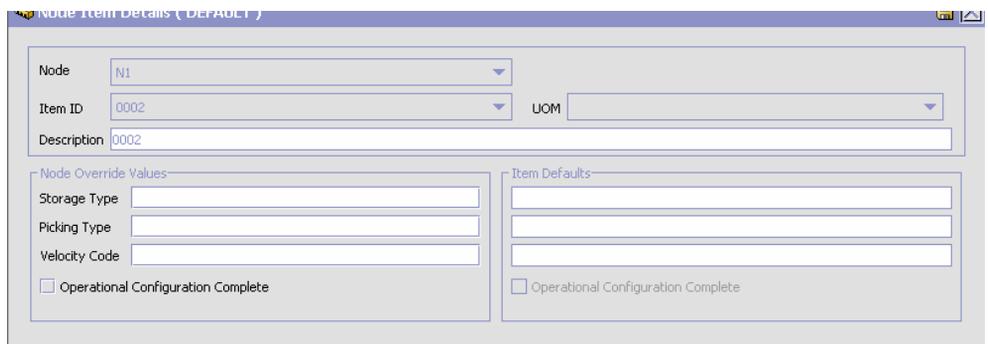


Node

Item ID

UOM

3. From Node, select the node that you want to define item attribute overrides for.
4. From Item ID, select the item whose attributes you want to define overrides for at the node level.
5. From UOM, select the unit of measure of the item ID you have selected to identify the specific item ID/unit of measure combination you want to define overrides for.
6. Choose . The Node Item Details window displays in the work area.
7. In the Node Override Values column, enter the override values for the applicable attributes. For attribute definitions, see [Section 4.1.2.6, "Defining a Product Item's Classifications"](#).
8. Choose .



Node Item Details (DEFAULT)

Node

Item ID UOM

Description

Node Override Values

Storage Type

Picking Type

Velocity Code

Operational Configuration Complete

Item Defaults

Operational Configuration Complete

4.2.2 Modifying Item Attribute Overrides at the Node Level

To modify item attribute overrides at the node level:

1. From the tree in the application rules side panel, choose Products > Node Item Attributes. The Node Items Search window displays in the work area.
2. Enter applicable search criteria and choose . The a list of items associated with the node you searched on displays.
3. Select the applicable item and choose . The Node Item Details window displays in the work area.
4. In the Node Override Values column, enter the override values for the applicable attributes. For attribute definitions, see [Section 4.1.2.6, "Defining a Product Item's Classifications"](#).
5. Choose .

4.2.3 Deleting Item Attribute Overrides at the Node Level

To delete item attribute overrides at the node level:

1. From the tree in the application rules side panel, choose Products > Node Item Attributes. The Node Items Search window displays in the work area.
2. Enter applicable search criteria and choose . The a list of items associated with the node you searched on displays.
3. Select the applicable item and choose .

4.3 Defining Master Units of Measure

You can define a master list of both quantity and pricing units of measure to be used when creating product items.

You can use the Item UOM Master branch for:

- [Defining Quantity Units of Measure](#)
- [Defining Pricing Units of Measure](#)

4.3.1 Defining Quantity Units of Measure

You can define a master list of quantity units of measure that can be used when defining unique item ID/unit of measure combinations and alternate ordering units of measure. For more information about defining a product item's unique unit of measure and alternate units of measure, see [Section 4.1, "Defining Product Items"](#).

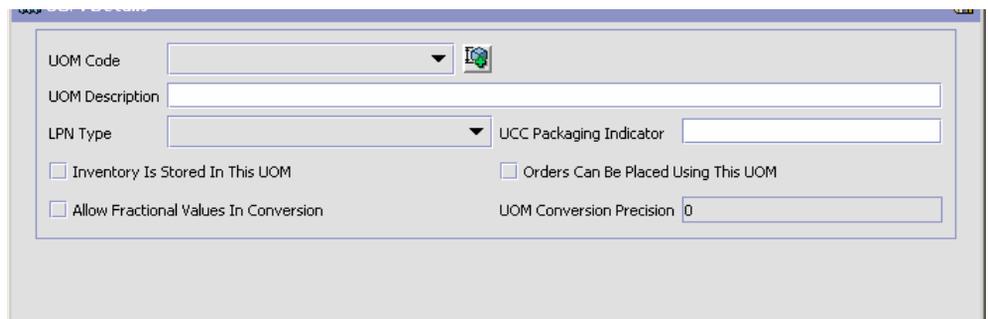
You can use the Quantity UOMs tab for:

- [Creating a Quantity Unit of Measure](#)
- [Modifying a Quantity Unit of Measure](#)
- [Deleting a Quantity Unit of Measure](#)

4.3.1.1 Creating a Quantity Unit of Measure

To create a quantity unit of measure:

1. From the tree in the application rules side panel, choose Products > Item UOM Master. The Product Item UOM Master window displays in the work area.
2. Choose the Quantity UOMs tab.
3. Choose . The UOM Details pop-up window displays in the work area.
4. Enter information in the applicable fields. Refer to [Table 4–26](#) for field value descriptions.
5. Choose .



The screenshot shows a window titled 'UOM Details' with the following fields and options:

- UOM Code:** A dropdown menu with a small icon to its right.
- UOM Description:** A text input field.
- LPN Type:** A dropdown menu.
- UCC Packaging Indicator:** A text input field.
- Inventory Is Stored In This UOM:** An unchecked checkbox.
- Orders Can Be Placed Using This UOM:** An unchecked checkbox.
- Allow Fractional Values In Conversion:** An unchecked checkbox.
- UOM Conversion Precision:** A text input field containing the value '0'.

Table 4–26 UOM Details Pop-Up Window

Field	Description
UOM Code	Select the unit of measure.
UOM Description	Enter a brief description of the unit of measure.
LPN Type	If the unit of measure represents a physical case, select Case. If the unit of measure represents a physical pallet, select Pallet.
UCC Packaging Indicator	Enter the UCC case code to be used as a standard for this all items associated with this unit of measure.
Inventory Is Stored In This UOM	Select Inventory Is Stored In This UOM if this unit of measure can be used when defining unique item ID/unit of measure combinations.
Orders Can Be Placed Using This UOM	Select Orders Can Be Placed Using This UOM if this unit of measure can be used when associating alternate ordering units of measure with a particular item ID/unit of measure combination.

Table 4–26 UOM Details Pop-Up Window

Field	Description
Allow Fractional Values in Conversion	<p>Select Allow Fractional Values in Conversion if you want to allow quantities of a fractional value when converting quantity from the inventory unit of measure to the alternate ordering unit of measure.</p> <p>Otherwise, it is assumed that the ordering unit of measure is always a positive integer multiple of the inventory unit of measure.</p> <p>For example, an item has an inventory unit of measure of EACH and an ordering unit of measure of Case. twelve EACH makes up one case. If you do not allow fractional values for the Case unit of measure you cannot ship ten units of the item and call it a Case.</p>
UOM Conversion Precision	<p>If you selected Allow Fractional Values in Conversion, the number of decimal places you want fractions to be rounded up to. For example, if you enter 2, the number 1.456 is rounded up to 1.46.</p> <p>If you are configuring a time unit of measure, the time is rounded up to the number of minutes you specify in this field. For example, if you set the conversion precision to 20 for a time unit of measure, something that takes 14 minutes would be rounded to 20 minutes and something that takes 67 minutes would be rounded up to 80 minutes.</p> <p>Note: The number is always rounded up to the specified precision.</p>

4.3.1.2 Modifying a Quantity Unit of Measure

To modify a quantity unit of measure:

1. From the tree in the application rules side panel, choose Products > Item UOM Master. The Product Item UOM Master window displays in the work area.
2. Choose the Quantity UOMs tab.
3. Select the applicable unit of measure and choose . The UOM Details pop-up window displays in the work area.
4. Enter information in the applicable fields. Refer to [Table 4–26](#) for field value descriptions.
5. Choose .

4.3.1.3 Deleting a Quantity Unit of Measure

To delete a quantity unit of measure:

1. From the tree in the application rules side panel, choose Products > Item UOM Master. The Product Item UOM Master window displays in the work area.
2. Choose the Quantity UOMs tab.
3. Select the applicable unit of measure and choose .

4.3.2 Defining Pricing Units of Measure

You can define a master list of pricing units of measure that can be used when a product item's pricing is based on a different unit of measure than it's inventory unit of measure.

For example, you have product item called Apples. Apples are inventoried and sold as a unit of measure called Cases. However, the pricing is determined by the weight of each case. In this scenario, you would assign the product item Apples the inventory unit of measure Cases and the pricing unit of measure Weight.

You can use the Pricing UOMs tab for:

- [Creating a Pricing Unit of Measure](#)
- [Modifying a Pricing Unit of Measure](#)
- [Deleting a Pricing Unit of Measure](#)

4.3.2.1 Creating a Pricing Unit of Measure

To create a pricing unit of measure:

1. From the tree in the application rules side panel, choose Products > Item UOM Master. The Product Item UOM Master window displays in the work area.
2. Choose the Pricing UOMs tab.
3. Choose . The UOM Details pop-up window displays in the work area.

The screenshot shows a window with three input fields:

- UOM Type:** A dropdown menu with "Weight" selected.
- UOM Code:** A dropdown menu that is currently empty, with a small icon to its right.
- UOM Description:** A text input field that is currently empty.

4. From UOM Type, select the applicable unit of measure type.
5. In UOM Code, select the unit of measure you want to be able to be used for pricing.
6. In UOM Description, enter a brief description of the pricing unit of measure.
7. Choose .

4.3.2.2 Modifying a Pricing Unit of Measure

To modify a pricing unit of measure:

1. From the tree in the application rules side panel, choose Products > Item UOM Master. The Product Item UOM Master window displays in the work area.
2. Choose the Pricing UOMs tab.
3. Select the applicable unit of measure and choose . The UOM Details pop-up window displays in the work area.
4. In UOM Description, enter a brief description of the pricing unit of measure.
5. Choose .

4.3.2.3 Deleting a Pricing Unit of Measure

To delete a pricing unit of measure:

1. From the tree in the application rules side panel, choose Products > Item UOM Master. The Product Item UOM Master window displays in the work area.

2. Choose the Pricing UOMs tab.
3. Select the applicable unit of measure and choose .

4.4 Defining Master Catalogs

The master catalog is the exhaustive list of your organization's items. You can have more than one master catalog if your business practices require different groupings for your master list of items.

Note: The Master Catalog Detail pop-up window is deprecated in this release and will be removed in a future release. See the *Item Concepts Guide* for information about the master catalog and selling catalogs, which are replacing the master catalog functionality described in [Defining Master Catalogs](#).

For example, you are configuring a retail organization that has two distinct products, shoes and shirts. Your business practice requires you to create two separate catalogs for the shoe line and the shirt line. You can set up a master catalog for all of the items that are included in the shoe line and a master catalog for all of the items that are included in the shirt line.

You can use the Master Catalog branch for:

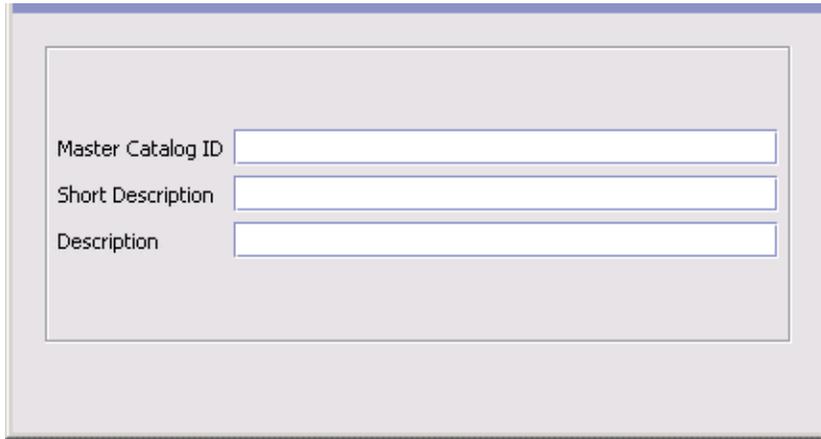
- [Creating a Master Catalog](#)
- [Modifying a Master Catalog](#)
- [Deleting a Master Catalog](#)

4.4.1 Creating a Master Catalog

To create a master catalog:

1. From the tree in the application rules side panel, choose Products > Master Catalog. The Master Catalog Search window displays in the work area.
2. Choose . The Master Catalog Detail pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 4-27](#) for field value descriptions.

4. Choose .



The screenshot shows a pop-up window with a light gray background. Inside, there is a white rectangular area containing three text input fields. The first field is labeled 'Master Catalog ID', the second is labeled 'Short Description', and the third is labeled 'Description'. Each field has a thin blue border and is currently empty.

Table 4–27 Master Catalog Detail Pop-Up Window

Field	Description
Master Catalog ID	Enter the master catalog's ID.
Short Description	Enter a brief description of the master catalog.
Description	Enter a more detailed description of the master catalog.

4.4.2 Modifying a Master Catalog

As inventory changes you may periodically need to update or modify your catalog.

To modify a master catalog:

1. From the tree in the application rules side panel, choose Products > Master Catalog. The Master Catalog Search window displays in the work area.
2. Enter applicable search information and choose . The Master Catalog list displays.
3. Select the applicable master catalog and choose . The Master Catalog Detail pop-up window displays.

4. Modify information in the applicable fields. Refer to [Table 4–27](#) for field value descriptions.
5. Choose .

4.4.3 Deleting a Master Catalog

To delete a master catalog:

1. From the tree in the application rules side panel, choose Products > Master Catalog. The Master Catalog Search window displays in the work area.
2. Enter applicable search information and choose . The Master Catalog list displays.
3. Select the applicable master catalog and choose .

4.5 Defining Categories

Categories provide your catalog with a means to describe the entire item set in a number of different hierarchal and searchable groupings. Each category can contain items from multiple catalogs, providing a single face to the users for all catalogs.

The highest level of the category hierarchy is called the catalog. All of the groupings that exist below the catalog are referred to as categories.

You can use the Categories branch for:

- [Defining Additional Category Attributes](#)

4.5.1 Defining Additional Category Attributes

Categories provide you with a means to describe the entire item set in a number of different hierarchal and searchable groupings.

You can use the Additional Category Attributes branch for:

- [Defining Category Statuses](#)
- [Defining Additional Attributes](#)

4.5.1.1 Defining Category Statuses

You can define common codes for category statuses used when setting up a category in Catalog Management.

Following are the Selling and Fulfillment Foundation default category statuses:

- 2000 - Held
- 3000 - Published

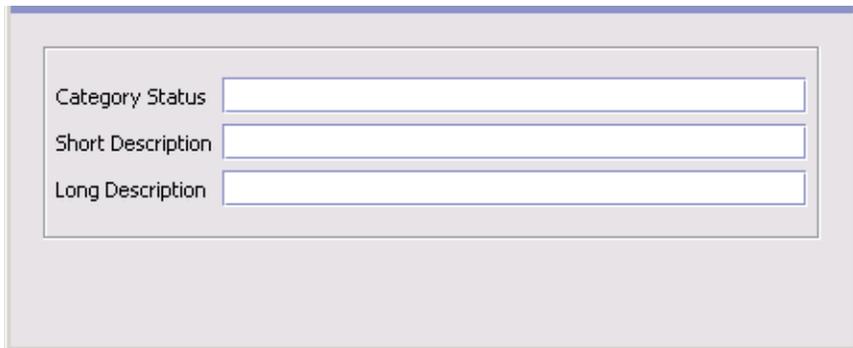
You can use the Category Statuses tab for:

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

4.5.1.1.1 Creating a Category Status

To create an category status:

1. From the tree in the application rules side panel, choose Products > Categories > Additional Category Attributes. The Catalog Categories window displays in the work area.
2. Choose the Category Statuses tab.
3. Choose . The Category Status Details pop-up window displays.



The screenshot shows a pop-up window titled 'Category Status Details'. It contains three text input fields:

- Category Status
- Short Description
- Long Description

4. In Category Status, enter the value you want to use for the category status.

5. In Short Description, enter a brief description of the category status.
6. In Long Description, enter a more detailed description of the category status.
7. Choose .

4.5.1.1.2 Modifying a Category Status

To modify an category status:

1. From the tree in the application rules side panel, choose Products > Categories > Additional Category Attributes. The Catalog Categories window displays in the work area.
2. Choose the Category Statuses tab.
3. Select the applicable category status and choose . The Category Status Details pop-up window displays.
4. In Short Description, enter a brief description of the category status.
5. In Long Description, enter a more detailed description of the category status.
6. Choose .

4.5.1.1.3 Deleting a Category Status

To delete an category status:

1. From the tree in the application rules side panel, choose Products > Categories > Additional Category Attributes. The Catalog Categories window displays in the work area.
2. Choose the Item Statuses tab.
3. Select the applicable category status and choose .

Note: You cannot delete default category statuses.

4.5.1.2 Defining Additional Attributes

You can define common codes for additional attributes used when setting up a product item. This allows you to store any additional information about your categories that is not captured by defaults in Selling and Fulfillment Foundation. These attributes appear as fields in the Category Details screen. You can create, modify, and delete additional attributes.

Following is the Selling and Fulfillment Foundation default additional attribute:

- DEFAULT

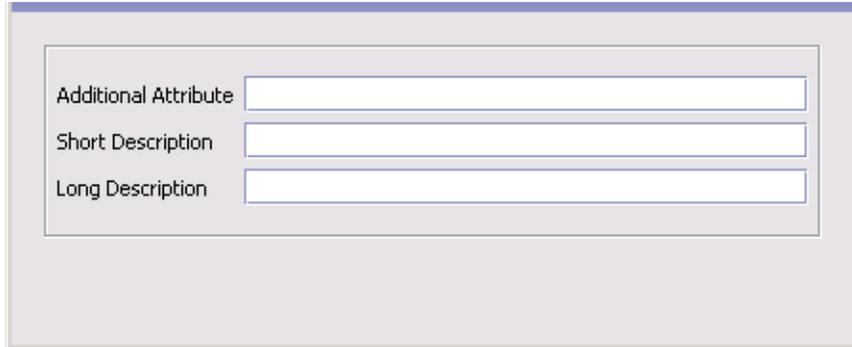
You can use the Additional Attributes tab for:

- [Creating an Additional Attribute](#)
- [Modifying an Additional Attribute](#)
- [Deleting an Additional Attribute](#)

4.5.1.2.1 Creating an Additional Attribute

To create an additional attribute:

1. From the tree in the application rules side panel, choose Products > Categories > Additional Category Attributes. The Catalog Categories window displays in the work area.
2. Choose the Additional Attributes tab.
3. Choose . The Additional Attribute Details pop-up window displays.



The screenshot shows a dialog box titled 'Additional Attribute Details'. It contains three text input fields:

- Additional Attribute
- Short Description
- Long Description

4. In Additional Attribute, enter the value you want to use for the additional attribute.

Important: Do not put a space or special characters in the Additional Attribute.

5. In Short Description, enter a brief description of the additional attribute.
6. In Long Description, enter a more detailed description of the additional attribute.
7. Choose .

4.5.1.2.2 Modifying an Additional Attribute

To modify an additional attribute:

1. From the tree in the application rules side panel, choose Products > Categories > Additional Category Attributes. The Catalog Categories window displays in the work area.
2. Choose the Additional Attributes tab.
3. Select the applicable additional attribute and choose . The Additional Attribute Details pop-up window displays.
4. In Short Description, enter a brief description of the additional attribute.
5. In Long Description, enter a more detailed description of the additional attribute.
6. Choose .

4.5.1.2.3 Deleting an Additional Attribute

To delete an additional attribute:

1. From the tree in the application rules side panel, choose Products > Categories > Additional Category Attributes. The Catalog Categories window displays in the work area.
2. Choose the Additional Attributes tab.
3. Select the applicable additional attribute and choose .

Note: You cannot delete the default additional attribute provided by Selling and Fulfillment Foundation.

4.6 Defining Classifications

You can define product item classifications and classification hierarchies that can be used within Selling and Fulfillment Foundation for actions such as sourcing, associating services, determining shipping preferences, and so on. By defining a classification you identify an item attribute as having a specified use in Selling and Fulfillment Foundation.

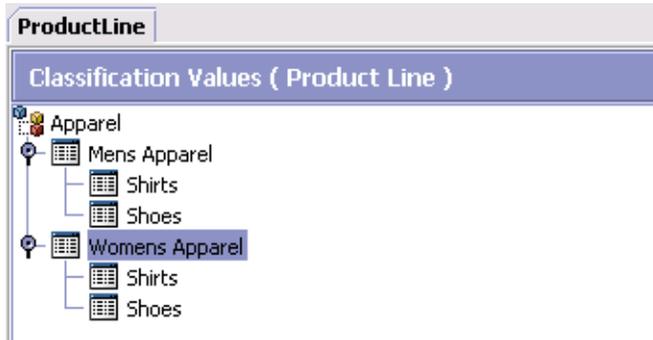
For example, you have items that contain hazardous materials and items that do not contain hazardous materials in your catalog. You want to source hazardous items out of Node 1 and non-hazardous materials out of Node 2. In this scenario, you can define a product item classification for the Hazardous Materials item attribute to be used for sourcing. You can then configure sourcing rules as needed for the Distributed Order Management application.

You can also define hierarchical item groupings for a classification. These groupings can be used to further refine the items affected by a classification purpose.

For example, you have created a classification called Apparel and associated it with the Product Line item attribute and Sourcing classification purpose. Under this classification you have two lines of clothing, Men's and Women's, each which contain items that are either classified as shirts or shoes. You want to source men's apparel from Node 1 and women's apparel from Node 2.

In this scenario, you can create a classification hierarchy for the Apparel classification as detailed in the following figure.

Figure 4–1 Classification Hierarchy Example



Once the hierarchy is created you can assign items to the applicable levels and configure sourcing rules to source all items associated with Men’s Apparel from Node 1 and all items associated with Women’s Apparel from Node 2.

A classification level overrides any classification level above it in the hierarchy. For example, continuing from the example above, you decide that you still want to source Men’s Apparel from Node 1. However, you want Womens Shoes to also be sourced from Node 1. In this scenario, if you configure the Shoes level underneath Womens Apparel to be sourced from Node 1, but still have the original configuration of all Womens Apparel being sourced from Node 2, the Shoes sourcing rule overrides the Womens Apparel sourcing rule.

For more information about configuring sourcing rules, see the *Sterling Distributed Order Management: Configuration Guide*.

You can use the Classifications branch for:

- [Defining Classification Definitions](#)
- [Defining Classification Purposes](#)
- [Defining a Classification Hierarchy](#)

4.6.1 Defining Classification Definitions

You can create a classification and associate an item attribute with it. Once a classification is defined, the corresponding field within the Product Item Details screen displays a lookup button from which the classification values you configure can be selected.

You can use the Classification Definitions tab for:

- [Creating a Classification Definition](#)
- [Modifying a Classification Definition](#)
- [Deleting a Classification Definition](#)

4.6.1.1 Creating a Classification Definition

To create a classification definition:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Definition. The Classification Definition window displays in the work area.
2. Choose the Classification Definitions tab.
3. Choose . The Classification Details pop-up window displays.

The screenshot shows a window titled 'Classification Definition'. It contains three input fields: 'Classification Name' (a text box), 'Item Attribute' (a dropdown menu), and 'Description' (a text box). The 'Item Attribute' dropdown menu is currently empty.

4. In Classification Name, enter the name of the classification.
5. From Item Attribute, select the item attribute you want to associate with the classification.

Note: The item attributes in the drop-down may display with underscores or without spacing. Disregard this.

6. In Description, enter a brief description of the classification.
7. Choose .

4.6.1.2 Modifying a Classification Definition

To modify a classification definition:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Definition. The Classification Definition window displays in the work area.
2. Choose the Classification Definitions tab.
3. Select the applicable classification definition and choose . The Classification Details pop-up window displays.
4. In Description, enter a brief description of the classification.
5. Choose .

4.6.1.3 Deleting a Classification Definition

To delete a classification definition:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Definition. The Classification Definition window displays in the work area.
2. Choose the Classification Definitions tab.
3. Select the applicable classification definition and choose .

Note: A classification definition binding to an item attribute Model must not be used to classify Item with Variations.

4.6.2 Defining Classification Purposes

Once you have created a classification definition, you can determine how you want to use it in the system. Refer to [Table 4–28](#) for the descriptions of default system classification purposes.

Table 4–28 Classification Purposes Definitions

Classification Purpose	Description
Capacity	Choosing 'Capacity' as a Classification Purpose enables one to define the same capacity constraints for a classification of items. For more information about Capacity, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
Procurement	Choosing 'Procurement' as a Classification Purpose enables a classification of items to drive specific procurement rule logic. For more information about Procurement, see the <i>Sterling Distributed Order Management: Configuration Guide</i> .
Sourcing 1 / Sourcing 2 / Sourcing 3	Choosing 'Sourcing' as a Classification Purpose enables a classification of items to drive specific sourcing rule logic. For more information about Sourcing, see the <i>Sterling Distributed Order Management: Configuration Guide</i> .
Container Category	Choosing 'Container Category' as a Classification Purpose enables one to containerize a classification of items in the same way. For more information about Container Category, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
Count Strategy 1 / Count Strategy 2 / Count Strategy 3	Choosing 'Count Strategy' as a Classification Purpose enables oneto use the same count strategy for a classification of items. For more information about Count Strategy, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
Distinct Attributes	Choosing Distinct Attributes as a Classification Purpose enables you to define a set of attributes that are unique for a given item.
Inherited Item Attributes	Choosing Inherited Item Attributes as a Classification Purpose enables you to define default values for item attributes based on classification.
Location Inventory Monitor 1 / Location Inventory Monitor 2 / Location Inventory Monitor 3	Choosing 'Location Inventory Monitor' as a Classification Purpose enables alerting locations when the inventory reaches specific inventory monitory levels, for a classification of items. For more information about Location Inventory Monitor, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .

Table 4–28 Classification Purposes Definitions

Classification Purpose	Description
NMFC	Choosing NMFC as a classification purpose enables one to group items according to the National Motor Freight Classification for accurate classification and pricing of shipments. For more information about NMFC, see Section 4.1.2.6, "Defining a Product Item's Classifications"
Pack Constraints 1 / Pack Constraints 2 / Pack Constraints 3	Choosing 'Pack Constraints' as a Classification Purpose enables one to define mix constraints for a classification of items. For more information about Pack Constraints, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
Productivity	Choosing 'Productivity' as a Classification Purpose enables one to categorize productivity metrics by classification of items. For more information about Productivity, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
Receiving Preference	Choosing 'Receiving Preference' as a Classification Purpose enables one to define the receiving rule for a classification of items. For more information about Receiving Preference, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
Retrieval 1 / Retrieval 2 / Retrieval 3	Choosing 'Retrieval' as a Classification Purpose enables one to define retrieval rules for a classification of items. For more information about Retrieval, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
Routing Guide	Choosing 'Routing Guide' as a Classification Purpose enables one to define routing rules for a classification of items. For more information about Routing Guide, see the <i>Sterling Logistics Management: Configuration Guide</i> .
Service Association	Choosing 'Service Association' as a Classification Purpose enables a classification of items to be associated with a set of provided or delivery services. For more information about service associations, see Section 4.6.3.6, "Associating Provided Services and Delivery Services to a Classification Value" .

Table 4–28 Classification Purposes Definitions

Classification Purpose	Description
Shipping Preference	Choosing 'Shipping Preference' as a Classification Purpose enables one to define shipping rules for a classification of items. For more information about Shipping Preference, see the <i>Sterling Supply Collaboration: Configuration Guide</i> .
SKU UNIT Size Code	Choosing 'SKU Unit Size Code' as a Classification Purpose enables one to define size codes for items, and hence define carton capacities for items with a specific size code. For more information about SKU Unit Size Code, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
Storage 1 / Storage 2 / Storage 3	Choosing 'Storage' as a Classification Purpose enables driving the putaway of a classification of items. For more information about Storage, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .
VAS	Choosing 'VAS' as a Classification Purpose enables driving value added services for a classification of items. For more information about Value Added Services, see the <i>Sterling Warehouse Management System: Configuration Guide</i> .

You can use the Classification Purposes tab for:

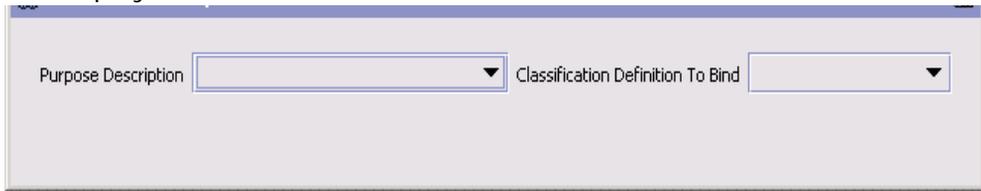
- [Creating a Classification Purpose](#)
- [Modifying a Classification Purpose](#)
- [Deleting Classification Purposes](#)

4.6.2.1 Creating a Classification Purpose

To create a classification purpose:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Definition. The Classification Definition window displays in the work area.
2. Choose the Classification Purposes tab.

3. Choose . The Classification Purpose Details pop-up window displays.



The screenshot shows a dialog box with two dropdown menus. The first dropdown is labeled 'Purpose Description' and the second is labeled 'Classification Definition To Bind'. Both dropdowns are currently empty.

4. From Purpose Description, select the applicable classification purpose.

Important: You cannot assign more than one classification definition to a classification purpose.

5. From Classification Definition To Bind, select the classification that you want to associate the classification purpose with.
6. Choose .

4.6.2.2 Modifying a Classification Purpose

To modify a classification purpose:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Definition. The Classification Definition window displays in the work area.
2. Choose the Classification Purposes tab.
3. Select the applicable classification purpose and choose . The Classification Purpose Details pop-up window displays.
4. From Classification Definition To Bind, select the classification that you want to associate the classification purpose with.
5. Choose .

4.6.2.3 Deleting Classification Purposes

To delete a classification purpose:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Definition. The Classification Definition window displays in the work area.

2. Choose the Classification Purposes tab.
3. Select the applicable classification purpose and choose .

4.6.3 Defining a Classification Hierarchy

A classification hierarchy contains all of the valid values a given classification can take in a hierarchical format. A classification is always associated to an attribute in the item master. This hierarchy stores all the values that the associated item attribute can take.

You can use the Classification Hierarchy branch for:

- [Creating a Classification Value](#)
- [Modifying and Adding Items to a Classification Value](#)
- [Defining Common Item Attributes for a Classification Value](#)
- [Deleting a Classification Value](#)

4.6.3.1 Creating a Classification Value

To create a classification value:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
2. Choose . The Value Details pop-up window displays.

The screenshot shows a software interface for defining classifications. It features a 'Value Details' window with three text input fields: 'Classification Value' (containing 'd'), 'Description' (containing 'd'), and 'Long Description' (containing 'd'). Below these fields are two tabs: 'Product Items' and 'Common Item Attributes'. The 'Product Items' tab is active, displaying a table with two columns: 'Item ID' and 'Unit Of Measure'. The table contains one row with 'Item001' in the 'Item ID' column. At the bottom of the table, it says 'Results 1 Of 1'.

Item ID	Unit Of Measure
Item001	

3. In Classification Value, enter the classification value.

Note: In instances where the Classification Definition is associated with the Item Attribute Velocity Code, only valid values (existing velocity codes) must be entered in the Value Details screen.

4. In Description, enter a brief description of the classification value.

Note: The description that you enter for the classification value is the literal that appears in the classification hierarchy on the user interface. You can have the same description for different classification values, therefore it is possible to have two or more of the same value descriptions on the user interface. If this happens you can discern the classification value the description applies to by performing a mouse-over.

5. In Long Description, enter a more detailed description of the classification value.
6. Choose .

4.6.3.2 Modifying and Adding Items to a Classification Value

To modify a classification value:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
2. Select the applicable classification value and choose . The Value Details pop-up window displays.
3. In Description, enter a brief description of the classification value.
4. In Long Description, enter a more detailed description of the classification value.
5. From the Classification Item List, choose . The Item Search window displays.
6. Enter applicable search criteria and choose . A list of items displays.
7. Select the item you want to associate with this classification.
8. Choose .

4.6.3.3 Modifying and Adding Items to a Classification Value

To modify a classification value:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
2. Select the applicable classification value and choose . The Value Details pop-up window displays.
3. In Description, enter a brief description of the classification value.
4. In Long Description, enter a more detailed description of the classification value.
5. From the Classification Item List, choose . The Item Search window displays.
6. Enter applicable search criteria and choose . A list of items displays.
7. Select the item you want to associate with this classification.
8. Choose .

4.6.3.4 Defining Common Item Attributes for a Classification Value

You can define common item attributes at the classification level. Items with this classification level inherit values for these attributes, unless the values for these attributes have been specified at the item level. For more information about item details, see [Section 4.1, "Defining Product Items"](#).

Note: To configure common item attributes, you must assign a classification purpose to a classification value. For more information about classification purposes, see [Section 4.6.2, "Defining Classification Purposes"](#).

To define common item attributes for a classification value:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.

2. Select the applicable classification value and choose . The Value Details pop-up window displays. Select the Common Item Attributes Tab.
3. Enter information into the applicable fields. Refer to [Table 4–29](#) for field level descriptions.
4. Choose .
- 5.

Product Items Common Item Attributes

Common Item Attributes

Reverse Logistics

Returnable Return Window Days

Inventory Info

Default Product Class	<input type="text"/>		ATP Rule	<input type="text"/>	
Lead Days	<input type="text"/>		ATP Monitor Rule	<input type="text"/>	
Track FIFO in Inventory	<input type="text"/>		Node Level ATP Monitor Rule	<input type="text"/>	
Future Safety Factor Percentage	<input type="text"/>				
<input checked="" type="radio"/> Onhand Safety Factor Percentage	<input type="text"/>	<input type="radio"/>	Onhand Safety Factor Quantity	<input type="text"/>	
Item-Based Allocation Allowed	<input type="text"/>		Use Unplanned Inventory	<input type="text"/>	

Sourcing Info

Release an order for this item a total of working hours and system days before expected time of shipment

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

Item Shipment Consolidation Window Days

Pickup Allowed	<input type="text"/>	Delivery Allowed	<input type="text"/>
Shipping Allowed	<input type="text"/>	Don't Perform Inventory Check on Schedule and Release	<input type="text"/>
Forwarding Allowed	<input type="text"/>	Procurement Allowed	<input type="text"/>
Item Substitution On Order Allowed	<input type="text"/>		

Transportation Info

Parcel Shipping Allowed Requires Freezer

Shipping By Air Allowed

Table 4–29 Value Details, Common Item Attributes Tab

Field	Description
View Icons	
	Click this icon to open the Extended Attributes pop-up window. If you have specified additional common item attributes, you can define them for the classification here.
Reverse Logistics	
Returnable	Specify whether this item can be returned.
Return Window Days	Enter the number of days from receipt of the item that the Buyer can return the item.
Inventory Info	
Default Product Class	Select any of the configured product classes based on your business practices. For more information about product classes, see the <i>Selling and Fulfillment Foundation: Application Platform Configuration Guide</i> .
ATP Rule	Select an ATP Rule to use. The parameters defined in ATP rules are used to determine the available inventory. For more information about ATP rules, see the <i>Sterling Global Inventory Visibility: Configuration Guide</i> . Note: If you do not choose an ATP rule, the system's DEFAULT ATP rule is used.
Lead Days	Enter the amount of time (in days) needed to procure inventory and make the items in the classification available for shipping.
ATP Monitor Rule	Select an Inventory Monitoring Rule to use. The minimum inventory levels are based on the parameters set up for the selected ATP Monitoring Rule. For more information about ATP monitoring rules, see the <i>Sterling Global Inventory Visibility: Configuration Guide</i> .
Track FIFO in Inventory	Indicates that inventory is tracked using the First In First Out (FIFO) number. The FIFO number is automatically generated by the system.

Table 4–29 Value Details, Common Item Attributes Tab

Field	Description
Node Level ATP Monitor Rule	Select an Node Level Inventory Monitoring Rule to use. The minimum inventory levels for this item are based on the parameters set up for the selected ATP Monitoring Rule. For more information about ATP monitoring rules, see the <i>Sterling Global Inventory Visibility: Configuration Guide</i> .
Future Safety Factor Percentage	Enter the percentage of inventory that you want to exclude from future inventory availability.
Item Based Allocation Allowed	Select the appropriate option from the drop-down list to allow the item-based allocation for an item. When the 'Use Item Based Allocation' rule is enabled, item-based allocation is only applicable for the items and nodes that 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> .
Use Unplanned Inventory	Select the appropriate option to use unplanned inventory for this item.
Onhand Safety Factor Percentage	Select this option to enter the percentage of inventory that you want to exclude from on hand inventory availability. If this option is chosen, you cannot specify an Onhand Safety Factor Quantity.
Onhand Safety Factor Quantity	Select this option to enter the quantity of inventory that you want to exclude from on hand inventory availability. If this option is chosen, you cannot specify an Onhand Safety Factor Percentage.
Sourcing Info	
Release an order for this item a total of <number of hours> working hours and <number of days> system days before expected time of shipment	Enter the total number of working hours and system days an order for this item should be released before its expected to ship.

Table 4–29 Value Details, Common Item Attributes Tab

Field	Description
Node needs to be notified at least <number of hours> hours prior to expected time of shipment	Enter the minimum number of hours a node needs to be notified before the expected time of shipment.
Item Shipment Consolidation window <number of days> Days	Enter the number of days an item can be delayed after its availability date in order to consolidate its shipment with other items.
Pickup Allowed	Specify whether pickup is allowed.
Delivery Allowed	Specify whether delivery is allowed. Options are: <ul style="list-style-type: none"> • Y - Yes, delivery is allowed • M - Maybe, only if something else is being delivered • N - No, delivery is not allowed
Shipping Allowed	Specify whether shipping is allowed. Options are: <ul style="list-style-type: none"> • Y - Yes, shipping is allowed • M - Maybe, only if delivery is not possible • N - No, shipping is not allowed
Don't Perform Inventory Check on Schedule and Release	Specify whether inventory checks should be performed during the Schedule and Release process.
Forwarding Allowed	Specify whether forwarding is allowed.
Procurement Allowed	Specify whether procurement is allowed.
Item Supersession on Order Allowed	Specify whether supersession for the item is allowed.
Transportation Info	
Parcel Shipping Allowed	Specify whether parcel shipping of transfers is allowed.

Table 4–29 Value Details, Common Item Attributes Tab

Field	Description
Requires Freezer	Specify whether items require freezer storage during transportation.
Shipping By Air Allowed	Specify whether items can be shipped by air.

4.6.3.5 Deleting a Classification Value

To delete a classification value:

1. From the tree in the application rules side panel, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
2. Select the applicable classification value and choose .

4.6.3.6 Associating Provided Services and Delivery Services to a Classification Value

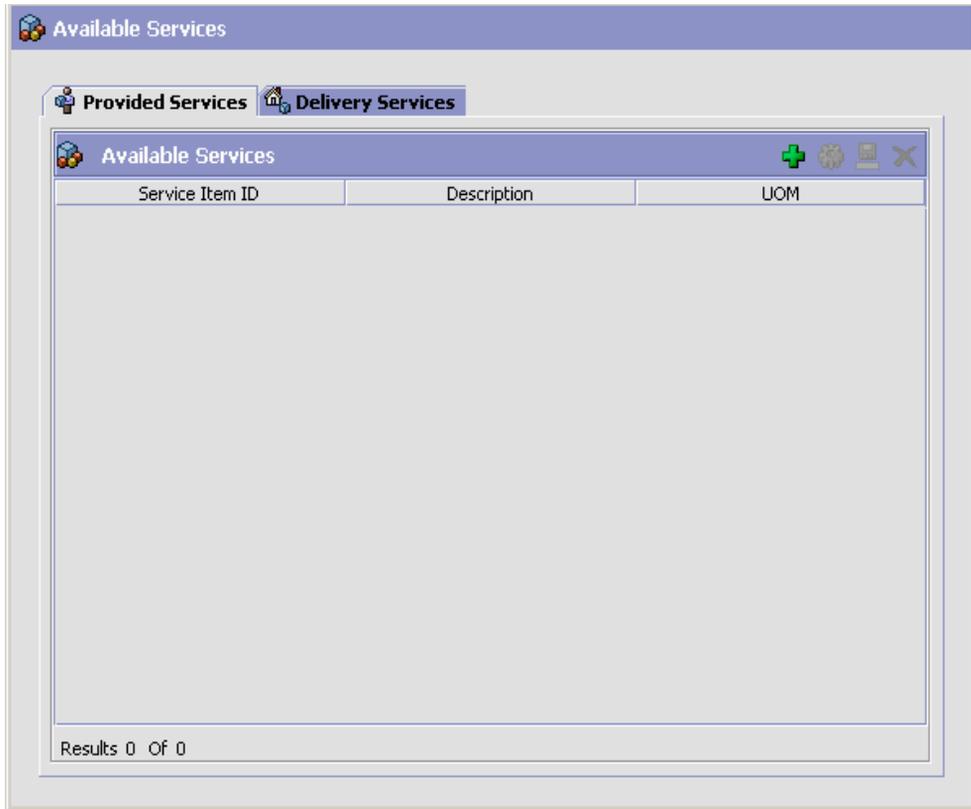
You can associate Services to a classification value.

You can use the Service Skills tab to:

- Associate provided services to a classification value
- Associate delivery services to a classification value

To associate services to a classification value:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
3. Choose . The Available Services pop-up window displays.



Use the provided services tab for:

- [Creating a Provided Service Association](#)
- [Modifying a Provided Service Association](#)
- [Deleting a Provided Service Association](#)

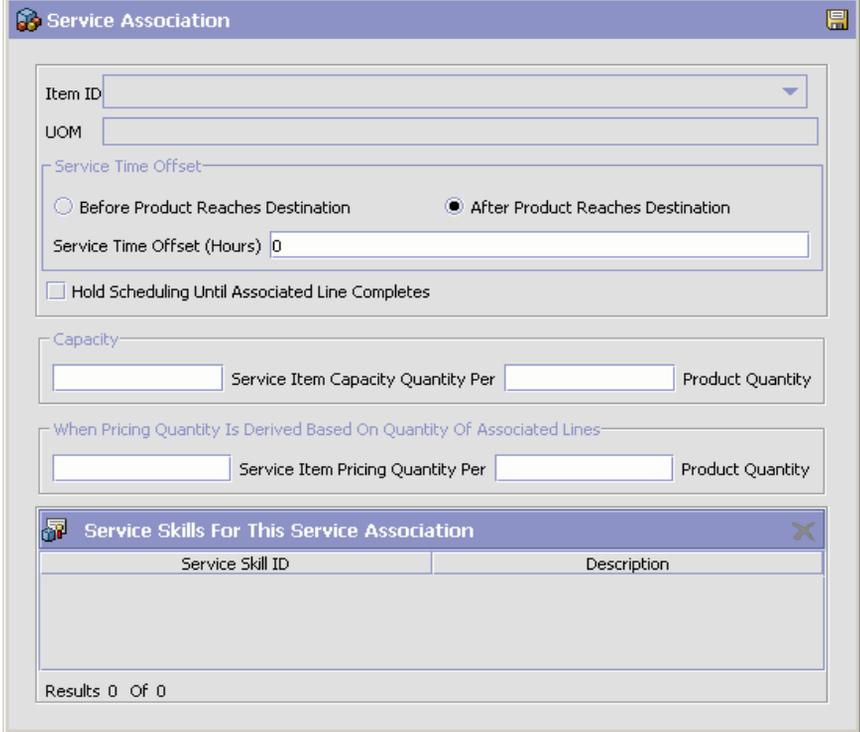
Use the delivery services tab for:

- [Creating a Delivery Service Association](#)
- [Modifying a Delivery Service Association](#)
- [Deleting a Delivery Service Association](#)

4.6.3.6.1 Creating a Provided Service Association

To create a provided service association:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
3. Choose . The Available Services pop-up window displays. Select the Provided services tab.
4. Choose . The Service Association Details pop-up window displays.



Service Association

Item ID

UOM

Service Time Offset

Before Product Reaches Destination After Product Reaches Destination

Service Time Offset (Hours)

Hold Scheduling Until Associated Line Completes

Capacity

Service Item Capacity Quantity Per Product Quantity

When Pricing Quantity Is Derived Based On Quantity Of Associated Lines

Service Item Pricing Quantity Per Product Quantity

Service Skills For This Service Association

Service Skill ID	Description

Results 0 Of 0

5. Enter information in the applicable fields. See [Table 4–13, "Service Association Pop-Up Window"](#) for field value descriptions.
6. Choose .

4.6.3.6.2 Modifying a Provided Service Association

To modify a provided service association:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
3. Choose . The Available Services pop-up window displays. Select the Provided services tab.
4. Choose . The Service Association Details pop-up window displays.
5. Enter information in the applicable fields. See [Table 4–13, "Service Association Pop-Up Window"](#) for field value descriptions.
6. Choose .

4.6.3.6.3 Deleting a Provided Service Association

To delete a provided service association:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
3. Choose . The Available Services pop-up window displays.
4. Select the Provided services tab.
5. Select the applicable provided service association and choose .

4.6.3.6.4 Creating a Delivery Service Association

To create a delivery service association:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.

3. Choose . The Available Services pop-up window displays. Select the Delivery services tab.
4. Choose . The Service Association Details pop-up window displays.
5. Enter information in the applicable fields. See [Table 4–13, "Service Association Pop-Up Window"](#) for field value descriptions.
6. Choose .

4.6.3.6.5 Modifying a Delivery Service Association

To modify a delivery service association:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
3. Choose . The Available Services pop-up window displays. Select the Delivery services tab.
4. Choose . The Service Association Details pop-up window displays.
5. Enter information in the applicable fields. See [Table 4–13, "Service Association Pop-Up Window"](#) for field value descriptions.
6. Choose .

4.6.3.6.6 Deleting a Delivery Service Association

To delete a delivery service association:

1. From the menu bar, choose Applications > Catalog Management. The Catalog Management tree displays in the side panel.
2. From the Catalog Management tree, choose Products > Classifications > Classification Hierarchy. The Classification Values window displays in the work area.
3. Choose . The Available Services pop-up window displays. Select the Delivery services tab.
4. Select the applicable delivery service association and choose .

4.7 Defining Item Attributes

You can define additional attributes that can be used when configuring a product item.

You can use the Item Attributes branch for:

- [Defining Alias Types](#)

4.7.1 Defining Alias Types

You can define common codes for alias types used when configuring a product item. Aliases appear as fields in the item details screen.

Following is the Selling and Fulfillment Foundation default alias type:

- UPC Code

You can use the Alias Types branch for:

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

4.7.1.1 Creating an Alias Type

To create an alias type:

1. From the tree in the application rules side panel, choose Products > Item Attributes > Alias Types. The Item Aliases window displays in the work area.
2. Choose . The Alias Type 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 'Item Alias Type', the second is labeled 'Short Description', and the third is labeled 'Long Description'. Each label is positioned to the left of its corresponding text box.

3. In Alias Type, enter the value you want to use for the alias type.

Important: Do not put a space or special characters in the Alias Type.

4. In Short Description, enter a brief description of the alias type.
5. In Long Description, enter a more detailed description of the alias type.
6. Choose .

4.7.1.2 Modifying an Alias Type

To modify an alias type:

1. From the tree in the application rules side panel, choose Products > Item Attributes > Alias Types. The Item Aliases window displays in the work area.
2. Select the applicable alias type and choose . The Alias Type Details pop-up window displays.
3. In Short Description, enter a brief description of the alias type.
4. In Long Description, enter a more detailed description of the alias type.
5. Choose .

4.7.1.3 Deleting an Alias Type

To delete an alias type:

1. From the tree in the application rules side panel, choose Products > Item Attributes > Alias Types. The Item Aliases window displays in the work area.
2. Select the applicable alias type and choose .

Note: You cannot delete default alias types.

4.7.2 Item Cost Posting Classification

You can group items into appropriate financial ledger groups by creating cost Posting Classifications when setting up an item in Catalog Management. When Selling and Fulfillment Foundation integrates with financial applications, the translation of a transaction from Selling and Fulfillment Foundation into a series of financial postings, and the classification value for the item in the transaction is carried into the financial application's interface.

4.7.2.1 Creating an Item's Cost Posting Classification

To create an item's cost posting classification:

1. From the tree in the application rules side panel, choose Products > Item Attributes > Cost Posting Classifications. The Cost Posting Classifications Types window displays in the work area.
2. Choose . The Cost Posting Classification Details pop-up window displays.

The screenshot shows a window titled 'Cost Posting Classification Details'. It contains three text input fields arranged vertically. The first field is labeled 'Cost Posting Classification', the second is labeled 'Short Description', and the third is labeled 'Long Description'. Each field is currently empty.

3. In Cost Posting Classification, enter the value you want to use for the item's financial ledger group classification.
4. In Short Description, enter a brief description of the cost posting classification.
5. In Long Description, enter a more detailed description of the cost posting classification.
6. Choose .

4.7.2.2 Modifying an Item's Cost Posting Classification

To modify an item's cost posting classification:

1. From the tree in the application rules side panel, choose Products > Item Attributes > Cost Posting Classifications. The Cost Posting Classifications window displays in the work area.
2. Select the applicable cost posting classification and choose . The Cost Posting Classification Details pop-up window displays.
3. In Short Description, enter a brief description of the cost posting classification.
4. In Long Description, enter a more detailed description of the cost posting classification.
5. Choose .

4.7.2.3 Deleting an Item's Cost Posting Classification

To delete an item's cost posting classification:

1. From the tree in the application rules side panel, choose Products > Item Attributes > Cost Posting Classifications. The Cost Posting Classifications window displays in the work area.
2. Select the applicable cost posting classification and choose .

5

Configuring Delivery Services

A delivery service is a service item that is used for last mile delivery associations with product items. Last mile delivery differs from order shipping in that specific product item deliveries are made using a carrier with defined capacity constraints. Delivery services are typically used for product items that are:

- Heavy, oversized, or fragile and cannot be transported by common carriers
- Require special handling which requires special equipment and/or personnel

Projection TVs and washing machines are examples of two product items that you might want to associate with a delivery service.

A delivery item is uniquely defined by its item ID and unit of measure. You can configure capacity to be maintained for delivery items as well as sourcing rules. For more information about configuring delivery service item capacity rules, see the *Sterling Global Inventory Visibility: Configuration Guide*. For more information about configuring delivery service item sourcing rules, see the *Sterling Distributed Order Management: Configuration Guide*.

You can use the Delivery Services branch for:

- [Defining Delivery Service Options](#)
- [Defining Delivery Service Master Units of Measure](#)
- [Defining Service Types](#)

5.1 Defining Delivery Service Options

You can define additional options that can be added to a delivery service. These options can add additional capacity requirements and cost to the delivery service you associate it with.

Note: Delivery service options can be associated with more than one delivery service item.

For example, you have configured a 2-Man Deliver and Hookup delivery service item that you use when delivering projection TVs. You can also create two delivery options to associate with the delivery service item called, Remove Trash and Haul Away. With these options a customer can decide if they want the delivery men to remove the trash left behind from the projection TV box, as well as whether they want the delivery men to haul off their old TV.

You can use the Delivery Service Options branch for:

- [Creating a Delivery Service Option](#)
- [Modifying a Delivery Service Option](#)
- [Deleting a Delivery Service Option](#)

5.1.1 Creating a Delivery Service Option

To create a delivery service option:

1. From the tree in the application rules side panel, choose Delivery Services > Delivery Service Options. The Delivery Service Option Search window displays in the work area.
2. Choose . The Option Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 5–1](#) for field value descriptions.
4. Choose .

The screenshot shows a pop-up window titled 'Option Details'. It has a light gray background and a white border. At the top, there are four input fields: 'Item ID' (text box with 'InstallOption'), 'UOM' (dropdown menu with 'Hour' selected), 'Short Description' (text box), and 'Fixed Capacity Units' (text box). Below these is a large, empty text area labeled 'Description'.

Table 5–1 Option Details Pop-Up Window

Field	Description
Item ID	Enter the delivery service option item ID.
UOM	Select the unit of measure to associate with the delivery service option. Important: You can only associate a delivery service option with a delivery service item that has the same unit of measure.
Short Description	Enter a brief description of the delivery service option.
Fixed Capacity Units	Enter the amount of capacity that is consumed by the delivery service based on its unit of measure. For example, if the unit of measure for this delivery service option is Hours and it takes one hour to perform fulfill the option, you would enter 1.
Description	Enter a detailed description of the delivery service option.

5.1.2 Modifying a Delivery Service Option

To modify a delivery service option:

1. From the tree in the application rules side panel, choose Delivery Services > Delivery Service Options. The Delivery Service Option Search window displays in the work area.
2. Enter applicable search criteria and choose . A list of delivery service options displays.
3. Select the applicable delivery service option and choose . The Option Details pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 5–1](#) for field value descriptions.
5. Choose .

5.1.3 Deleting a Delivery Service Option

To delete a delivery service option:

1. From the tree in the application rules side panel, choose Delivery Services > Delivery Service Options. The Delivery Service Option Search window displays in the work area.
2. Enter applicable search criteria and choose . A list of delivery service options displays.
3. Select the applicable delivery service option and choose .

5.2 Defining Delivery Service Master Units of Measure

You can define a master list of capacity and pricing units of measure to be used when creating a delivery service item.

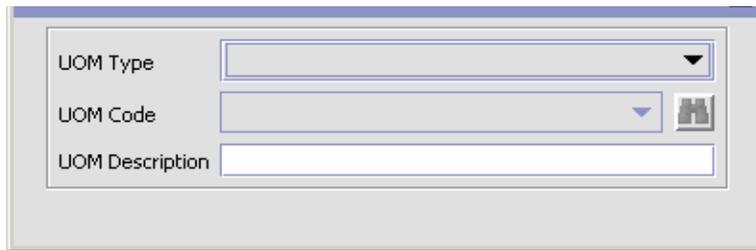
You can use the Delivery Service UOM Master branch for:

- [Creating a Master Unit of Measure](#)
- [Modifying a Master Unit of Measure](#)
- [Deleting a Master Unit of Measure](#)

5.2.1 Creating a Master Unit of Measure

To create a master unit of measure:

1. From the tree in the application rules side panel, choose Delivery Services > Delivery Service UOM Master. The Delivery Services Units of Measure window displays in the work area.
2. Choose . The UOM Details pop-up window displays.



3. From UOM Type, select the applicable unit of measure type.
4. In UOM Code, select the unit of measure you want to be able to be used for delivery services.
5. In UOM Description, enter a brief description of the delivery service unit of measure.
6. Choose .

5.2.2 Modifying a Master Unit of Measure

To modify a master unit of measure:

1. From the tree in the application rules side panel, choose Delivery Services > Delivery Service UOM Master. The Delivery Services Units of Measure window displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details pop-up window displays.
3. In UOM Description, enter a brief description of the delivery service unit of measure.
4. Choose .

5.2.3 Deleting a Master Unit of Measure

To delete a master unit of measure:

1. From the tree in the application rules side panel, choose Delivery Services > Delivery Service UOM Master. The Delivery Services Units of Measure window displays in the work area.
2. Select the applicable unit of measure and choose .

5.3 Defining Service Types

You can use the service types branch for:

- [Defining Delivery Service Types](#)
- [Defining Additional Capacity](#)

5.3.1 Defining Delivery Service Types

You can define service types to be used when creating a delivery service item.

You can use the Service Type branch for:

- [Creating a Delivery Service Type](#)
- [Modifying a Delivery Service Type](#)
- [Deleting a Delivery Service Type](#)

5.3.1.1 Creating a Delivery Service Type

To create a delivery service type:

1. From the Catalog Management tree in the application rules side panel, choose Delivery Services > Service Type > Delivery Service Types. The Delivery Service Types window displays in the work area.
2. Choose . The Service Type Details pop-up window displays.

Service Type ID	DS-ST	
Description	DS-ST	
Service Complexity Level	Simple	
Capacity UOM	Unit	
Service Item ID For Serviced Area Search	SimpleDelivery	
Service Item UOM For Serviced Area Search	Unit	

Table 5–2 Service Type Details Pop-up Window

Field	Descriptions
Service Type ID	Enter a unique identifier for the service type. The Service Type ID must be unique across delivery and provided services.
Description	Enter a description of the service type.
Service Complexity Level	Select a service complexity level. Note: Service Complexity Level is mandatory when defining a Service Type. Refer to Section 3.6, "Defining Service Complexity Levels" for more information.
Capacity UOM	Select a capacity UOM.
Service Item ID For Serviced Area Search	If you are planning on using the Serviced Area Search feature in the Console, select a service item identifier from the drop-down list that is representative of typical delivery services.
Service Item UOM For Service Area Search	If you are planning on using the Serviced Area Search feature in the PC prod-console , select a service item unit of measure from the drop-down list that is representative of typical delivery services.

3. Enter information in the applicable fields. Refer to [Table 5–2](#) for more information.
4. Choose .

5.3.1.2 Modifying a Delivery Service Type

To modify a delivery service type:

1. From the Catalog Management tree in the application rules side panel, choose Delivery Services > Service Type > Delivery Service Types. The Delivery Service Types window displays in the work area.
2. Select the applicable service type and choose . The Service Type Details pop-up window displays.
3. Edit information in the applicable fields. Refer to [Table 5–2](#) for more information.
4. Choose .

5.3.1.3 Deleting a Delivery Service Type

To delete a delivery service type:

1. From the Catalog Management tree in the application rules side panel, choose Delivery Services > Service Type > Delivery Service Types. The Delivery Service Types window displays in the work area.
2. Select the applicable service type and choose .

5.3.2 Defining Additional Capacity

Additional capacity can be defined for a given service type and region level combination. For example, deliveries may take longer in the suburb than in the city, therefore you may want to always add a certain amount of time per delivery for a certain region, for a certain service type. Selling and Fulfillment Foundation allows you to do this, through the Additional Capacity window.

You can use the Additional Capacity branch for:

- [Creating Additional Capacity](#)
- [Modifying Additional Capacity](#)
- [Deleting Additional Capacity](#)

5.3.2.1 Creating Additional Capacity

To create additional capacity:

1. From the Catalog Management tree in the application rules side panel, choose Delivery Services > Service Type > Additional Capacity. The Additional Capacity window displays in the work area.

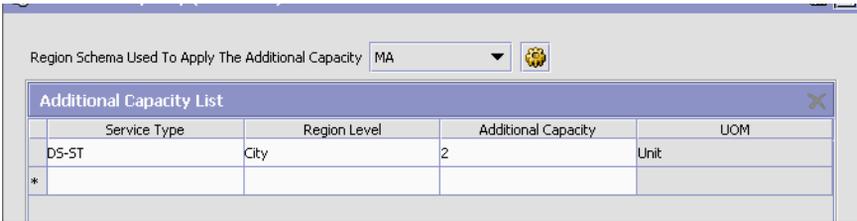


Table 5–3 Additional Capacity Window

Field	Description
Region Schema Used To Apply The Additional Capacity	From the drop-down list, select the region schema to which the additional capacity are applied.
Additional Capacity	
Service Type	From the drop-down list, select the service type for which you want to add additional capacity.
Region Level	From the drop-down list, select the region level for which you want to add the selected service type.
Additional Capacity	Double-click this field and enter the amount of extra capacity to add for the selected service type and region level.
UOM	The unit of measure associated with the selected service type. This field is automatically populated, and read-only.

2. Enter the information in the applicable fields. Refer to [Table 5–3](#) for field level description.
3. Click .

5.3.2.2 Modifying Additional Capacity

To modify additional capacity:

1. From the Catalog Management tree in the application rules side panel, choose Delivery Services > Service Type > Additional Capacity. The Additional Capacity window displays in the work area.
2. Enter the information in the applicable fields. Refer to [Table 5–3](#) for field level description.
3. Click .

5.3.2.3 Deleting Additional Capacity

To delete additional capacity:

1. From the Catalog Management tree in the application rules side panel, choose Delivery Services > Service Type > Additional Capacity. The Additional Capacity window displays in the work area.
2. Right click on the row that you want to remove from the additional capacity table and click Delete.
3. Click .

6

Configuring Provided Services

A provided service is any additional action that can be provided in an order that does not represent a physical product item. For example, you may have a Washing Machine product item in your catalog that you want to provide installation for. In this scenario, you can create an Installation provided service that can be offered when a customer orders the washing machine.

Note: Provided services do not have to be associated with a product item. You can configure a provided service to stand alone as its own service item.

A provided service item is uniquely defined by its item ID and unit of measure. You can configure capacity to be maintained for provided service items as well as sourcing rules. For more information about configuring provided service item capacity rules, see the *Sterling Global Inventory Visibility: Configuration Guide*. For more information about configuring provided service item sourcing rules, see the *Sterling Distributed Order Management: Configuration Guide*.

You can use the Provided Services branch for:

- [Defining Provided Service Options](#)
- [Defining Provided Service Master Units of Measure](#)
- [Defining Provided Service Types](#)

6.1 Defining Provided Service Options

You can define additional options that can be added to a provided service. These options can add additional capacity requirements and cost to the provided service you associate it with.

Note: Provided service options can be associated with more than one provided service item.

For example, you have configured an Installation service item that you use when installing carpet. You can also create a provided service option to associate with the delivery service item called Remove Trash. With this option a customer can decide if they want the carpet installers to remove the trash left behind from the carpet installation.

You can use the Provided Service Options branch for:

- [Creating a Provided Service Option](#)
- [Modifying a Provided Service Option](#)
- [Deleting a Provided Service Option](#)

6.1.1 Creating a Provided Service Option

To create a provided service option:

1. From the tree in the application rules side panel, choose Provided Services > Provided Service Options. The Provided Service Option Search window displays in the work area.
2. Choose . The Option Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 6–1](#) for field value descriptions.
4. Choose .

The screenshot shows a form titled 'Option Details Pop-Up Window'. It has a light gray background. At the top, there are four input fields: 'Item ID' containing 'InstallOption', 'UOM' with a dropdown menu set to 'Hour', 'Short Description' (empty), and 'Fixed Capacity Units' (empty). Below these is a large text area labeled 'Description' which is currently blank.

Table 6–1 Option Details Pop-Up Window

Field	Description
Item ID	Enter the provided service option item ID as you want it to appear throughout the system.
UOM	Select the unit of measure to associate with the provided service option. Important: You can only associate a provided service option with a provided service item that has the same unit of measure.
Short Description	Enter a brief description of the provided service option.
Fixed Capacity Units	Enter the amount of capacity that is consumed by the provided service based on its unit of measure. For example, if the unit of measure for this provided service option is Hours and it takes two hours to fulfill the option, you would enter 2.
Description	Enter a detailed description of the provided service option.

6.1.2 Modifying a Provided Service Option

To modify a provided service option:

1. From the tree in the application rules side panel, choose Provided Services > Provided Service Options. The Provided Service Option Search window displays in the work area.
2. Enter applicable search criteria and choose . A list of provided service options displays.
3. Select the applicable provided service option and choose . The Option Details pop-up window displays.
4. Enter information in the applicable fields. Refer to [Table 6–1](#) for field value descriptions.
5. Choose .

6.1.3 Deleting a Provided Service Option

To delete a provided service option:

1. From the tree in the application rules side panel, choose Provided Services > Provided Service Options. The Provided Service Option Search window displays in the work area.
2. Enter applicable search criteria and choose . A list of provided service options displays.
3. Select the applicable provided service option and choose .

6.2 Defining Provided Service Master Units of Measure

You can define a master list of capacity and pricing units of measure to be used when creating a provided service item.

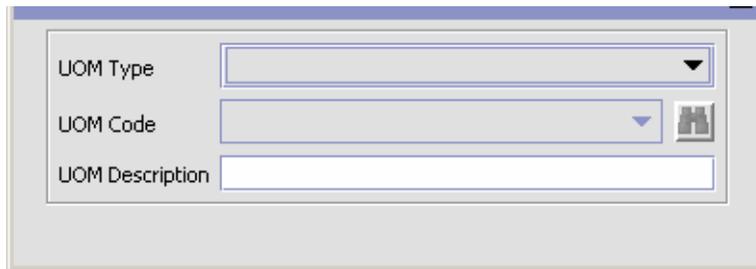
You can use the Provided Service UOM Master branch for:

- [Creating a Master Unit of Measure](#)
- [Modifying a Master Unit of Measure](#)
- [Deleting a Master Unit of Measure](#)

6.2.1 Creating a Master Unit of Measure

To create a master unit of measure:

1. From the tree in the application rules side panel, choose Provided Services > Provided Service UOM Master. The Provided Services Units of Measure window displays in the work area.
2. Choose . The UOM Details pop-up window displays.



3. From UOM Type, select the applicable unit of measure type.
4. In UOM Code, select the unit of measure you want to be able to be used for provided services.
5. In UOM Description, enter a brief description of the provided service unit of measure.
6. Choose .

6.2.2 Modifying a Master Unit of Measure

To modify a master unit of measure:

1. From the tree in the application rules side panel, choose Provided Services > Provided Service UOM Master. The Provided Services Units of Measure window displays in the work area.
2. Select the applicable unit of measure and choose . The UOM Details pop-up window displays.
3. In UOM Description, enter a brief description of the provided service unit of measure.

4. Choose .

6.2.3 Deleting a Master Unit of Measure

To delete a master unit of measure:

1. From the tree in the application rules side panel, choose Provided Services > Provided Service UOM Master. The Provided Services Units of Measure window displays in the work area.
2. Select the applicable unit of measure and choose .

6.3 Defining Provided Service Types

You can define service types to be used when creating a provided service item.

You can use the Provided Service Types branch for:

- [Creating a Provided Service Type](#)
- [Modifying a Provided Service Type](#)
- [Deleting a Provided Service Type](#)

6.3.1 Creating a Provided Service Type

To create a provided service type:

1. From the Catalog Management tree in the application rules side panel, choose Provided Services > Provided Service Types. The Provided Service Types window displays in the work area.
2. Choose . The Service Type Details pop-up window displays.

Service Type ID	DS-ST
Description	DS-ST
Service Complexity Level	Simple 
Capacity UOM	Unit 
Service Item ID For Serviced Area Search	SimpleDelivery
Service Item UOM For Serviced Area Search	Unit

Table 6–2 Service Type Details Pop-up Window

Field	Descriptions
Service Type ID	Enter a unique identifier for the service type. The Service Type ID must be unique across delivery and provided services.
Description	Enter a description of the service type.
Service Complexity Level	Select a service complexity level. Note: Service Complexity Level is mandatory when defining a Service Type. Refer to Section 3.6, "Defining Service Complexity Levels" for more information.
Capacity UOM	Select a capacity UOM.
Service Item ID For Serviced Area Search	If you are planning on using the Serviced Area Search feature in the PC prod-console , select a service item identifier from the drop-down list that is representative of typical provided services.
Service Item UOM For Service Area Search	If you are planning on using the Serviced Area Search feature in the PC prod-console , select a service item unit of measure from the drop-down list that is representative of typical provided services.

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

6.3.2 Modifying a Provided Service Type

To modify a provided service type:

1. From the Catalog Management tree in the application rules side panel, choose Provided Services > Provided Service Types. The Provided Service Types window displays in the work area.
2. Select the applicable service type and choose . The Service Type Details pop-up window displays.
3. Edit information in the applicable fields. Refer to [Table 6–2](#) for more information.
4. Choose .

6.3.3 Deleting a Provided Service Type

To delete a provided service type:

1. From the Catalog Management tree in the application rules side panel, choose Provided Services > Provided Service Types. The Provided Service Types window displays in the work area.
2. Select the applicable service type and choose .

Configuring Value Added Services

Value-Added Services (VAS) are performed to meet customer demands. These can be the demands of a buyer, such as special creating a customized order, or doing special processes for the buyer at the warehouse, or can be provided services to the customer, such as installing a light fixture.

When a work order is created manually, it consists of one or more services. These services can be of the following types:

- Kitting Service - Describes assembling components for an item
- DeKitting Service - Describes disassembling an item. This may be done to acquire an item component to complete another order.
- Compliance Service - Describes value added services that should be performed to supply an item to a specific buyer. Used in for Made-to-customer orders.
- Inventory Change Service - Describes how to convert inventory from one UOM to another.

One or more services can be included in work order, and have a sequence number assigned which indicates the order in which they should be performed. The activities are performed at all types of facilities including manufacturing facilities, flow-through and distribution centers.

Each service can contain one or more service activity, which specifies the category of activity, such as Assemble Components, Apply Logos, or Pack Components.

In addition, Provided Services can generate work orders. Provided Services represent services that can be ordered, such as item installation.

For more information about value added services and how they can be used, see the *Selling and Fulfillment Foundation: Product Concepts Guide*.

Note: If you are running a Sterling Warehouse Management System deployment, use Catalog Management to configure master data, such as value added services. See the appropriate section in this guide for configuration instructions. However, if you are running a Sterling Distributed Order Management deployment, use the Business Center to manage master data. See the *Business Center: Item Administration Guide* for instructions on using the Business Center application.

This chapter describes:

- [Defining a Value Added Service](#)
- [Defining an Activity](#)

7.1 Defining a Value Added Service

There are five value added services. They are:

- Kitting Services
- Dekitting Services
- Compliance Services
- Inventory Change Services
- Provided Services

All of the value added services are configured similarly, except for Provided Services. The activities for the first four value added services are:

- [Creating a Value Added Service](#)
- [Modifying a Value Added Service](#)
- [Deleting a Value Added Service](#)

7.1.1 Creating a Value Added Service

To create a value added service for Kitting Services, DeKitting Services, Compliance Services, or Inventory Change Services:

1. From the tree in the application rules side panel, choose the value added service you wish to add, such as Kitting Service.
2. A window with a list of the services of that type displays.
3. Choose . The value added services window for the selected service displays.
4. Enter an Item ID.
5. Choose .

You can use the Provided Service Details window for:

- [Defining a Value Added Service's Primary Information](#)
- [Defining a Value Added Service's Item Instructions](#)
- [Defining a Value Added Service's Service Activities](#)

7.1.1.1 Defining a Value Added Service's Primary Information

To define a value added service's primary information:

1. In the Value Added Service's Detail window, select the Primary Info tab.
2. Enter information in the applicable fields. Refer to [Table 7-1, "Primary Info Tab"](#) for field value descriptions.

The screenshot shows a software window with three tabs: 'Primary Info', 'Item Instructions', and 'Service Activities'. The 'Primary Info' tab is selected. It contains the following elements:

- Short Description:** A text input field.
- Description:** A text input field.
- Status:** A dropdown menu with a small icon to its right.
- Extended Description:** A large, empty text area with a 'Localize...' button in the top right corner.

Table 7–1 Primary Info Tab

Field	Description
Short Description	Enter a description which summarizes this Value Added Service
Description	Enter a more detailed description of the Value Added Service.
Status	The status for this value added service record. Select Held (Unpublished) to defer making this value added service available for use. Otherwise, select Published to make this value added service available in the system.

Table 7-1 Primary Info Tab

Field	Description
<p>Create Work Order While Scheduling An Order Requiring this Service</p>	<p>Used only for <i>compliance services</i>.</p> <p>Select this field to automatically generate a work order as necessary during scheduling an order. This is typically used when the scheduling needs to consider setting aside items to supply the order, or when there is a need for allocating time to perform the compliance service.</p> <p>When scheduling processing is performed, the processing may detect the need create more items to meet the needs of an order.</p> <p>This field lets you control whether a work order should be generated, based on your procedures.</p> <p>For example, you may have a compliance service that consists of adding a warranty card to each telephone shipped. Because of the way that the telephones are packed, this can be done as part of the packing process, and no separate work order is required. There is also no need to separately reserve the telephones for this particular order, as they do not to have any other special handling, and could be used to supply other order demands. In such a situation, you would not select this field.</p> <p>In a different scenario, the buyer's corporate label is applied to the telephones. In order to schedule the order, a work order is generated to track the process, and inventory may be allocated to create the customized telephones. These customized telephones could not be used to supply another buyer's order. In this case, you would select this field, to automatically generate the work order and to allocate stock to fulfil the compliance service task.</p>

Table 7–1 Primary Info Tab

Field	Description
Run Quantity	<p>Used only for <i>kitting services</i>.</p> <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 are 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>
Extended Description	<p>Enter a more detailed description. This can be localized by choosing the Localize button.</p>

7.1.1.2 Defining a Value Added Service's Item Instructions

You can define any special instructions that are appropriate to an item. For example, you can associate a special instruction of Handle With Care with a fragile product item.

You can use the Item Instructions tab for:

- [Creating Item Instructions](#)
- [Modifying an Item Instruction](#)
- [Deleting an Item Instruction](#)

7.1.1.2.1 Creating Item Instructions

To add an item instruction:

1. In the Value Added Services window, choose the Item Instructions tab.
2. From the Item Instructions table, choose . The Item Instruction Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 7–2](#) for field value descriptions.
4. Choose .

Table 7–2 Item Instruction Details Pop-Up Window

Field	Description
Item Instruction Type	Select the item instruction type. For example, a Handling item type indicates the instruction pertains to handling the product item.
Sequence No	If there are multiple, sequential instructions for the product item, enter the sequence of the instruction you are adding.
Use Item Instruction Code	Choose Use Item Instruction Code if you want to use a pre-existing item instruction code. For example, you could have a pre-existing Handle With Care item instruction code that you can associate with fragile items.
Item Instruction Code	If you chose Use Item Instruction Code, select the applicable item instruction code.

Table 7–2 Item Instruction Details Pop-Up Window

Field	Description
Create New Item Instruction	Select Create New Item Instruction if no pre-existing item instruction code exists and you want to create a free-form text instruction.
Text	If you selected Create New Item Instruction, enter the instruction.

7.1.1.2.2 Modifying an Item Instruction

To modify a item instruction:

1. In the Value Added Services window, choose the Item Instructions tab.
2. From the Item Instructions table, select the applicable item instruction choose . The Item Instruction Details pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 7–2](#) for field value descriptions.
4. Choose .

7.1.1.2.3 Deleting an Item Instruction

To delete an item instruction:

1. In the Value Added Services window, choose the Item Instructions tab.
2. From the Item Instructions table, select the applicable item instruction choose .

7.1.1.3 Defining a Value Added Service's Service Activities

The Service activities are the high level steps in providing a service. For example, if the value added service consisted of assembling a telephone, the service activities might be:

- Getting components
- Adding accessories
- Applying custom label

You can use this window for:

[Creating a Service Activity](#)

[Modifying a Service Activity](#)

[Deleting a Service Activity](#)

7.1.1.3.1 Creating a Service Activity

To create a service activity:

1. In the Value Added Services window, choose the Service Activity tab.
2. Choose . The Service Activity pop-up window displays.
3. Select an Activity Code and a sequence number, which is used to order the actions for the service.
4. Choose .

Note: At least one Service Activity should be associated with a Value-Added Service Item for the inventory updates to occur using the `confirmWorkOrder()` or `confirmWorkOrderActivity()` API.

You can also use this window for:

- [Adding a Service Activity Instructions](#)
- [Modifying a Service Activity Instruction](#)
- [Deleting a Service Activity Instruction](#)

Adding a Service Activity Instructions

Service Activity Instructions provide details on how to perform a service activity. For example, the service activity may be "Assemble product". The Service Activity instructions provides the details for how to do this, such as where coupons should be place in a product package, how special labelling should be applied, or other detailed instructions.

To add a Service Activity Instruction:

1. From the Service Activity Details pop-up window, choose .

- 2. The Service Activity Instructions Details window displays.
- 3. Enter information in the applicable fields. Refer to [Table 7-3](#) for field value descriptions.

The screenshot shows a window titled 'Service Activity Instructions Details'. It contains the following fields:

- Sequence No:** A text box containing the number '2'.
- Instruction URL:** An empty text box.
- Instruction Usage:** An empty text box.
- Text:** A large, empty text area for entering details.

Table 7-3 Service Activity Instruction Details Window

Field	Description
Sequence No	The sequence no is used to order the activity instructions. For example, a service activity instruction with a sequence no of '1' should be performed before a service activity instruction of '2'.
Instruction URL	Provides a URL which points to existing instructions for this service. For example, you may already have a existing procedure for inserting promotional materials into certain packages. Enter the URL for that procedure here.
Instruction Usage	Any comments on how this instruction should be used
Text	Enter details of the instruction.

Modifying a Service Activity Instruction

To modify a service activity:

4. From the Service Activity table, select the applicable service activity and choose . The Service Activity Details pop-up window displays.
5. In the Service Activity Instructions list, choose . The Service Activity Instructions Details window displays.
6. Enter information in the applicable fields. Refer to [Table 7–2](#) for field value descriptions.
7. Choose .

Deleting a Service Activity Instruction

To delete a service activity instruction:

1. In the Value Added Services window, choose the Service Activity tab.
2. From the Service Activity table, select the applicable service activity and choose .
3. From the Item Instructions table, select the applicable service activity and choose .

7.1.1.3.2 Modifying a Service Activity

To modify a service activity:

1. In the Value Added Services window, choose the Service Activity tab.
2. Select a Service Activity, and choose . The Service Activity pop-up window displays.
3. Enter information in the applicable fields. Refer to [Table 7–2](#) for field value descriptions.
4. Choose .

7.1.1.3.3 Deleting a Service Activity

To delete a service activity:

1. In the Value Added Services window, choose the Service Activity tab.
2. Select a Service Activity, and choose .

7.1.2 Modifying a Value Added Service

To modify a value added service:

1. From the tree in the application rules side panel, choose the value added service you wish to modify, such as Kitting Service.
2. A Value window with a list of the services of that type displays.
3. Select the value added service you wish to modify, and choose . The Value Added Services detail window displays.
4. Enter information in the applicable fields.
5. Choose .

7.1.3 Deleting a Value Added Service

1. From the tree in the application rules side panel, choose the value added service you wish to delete, such as Kitting Service.
2. A Value window with a list of the services of that type displays.
3. Select the value added service you wish to delete, and choose .

7.2 Defining an Activity

Activities define the categories of service activity in Selling and Fulfillment Foundation.

This section describes:

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

7.2.1 Creating an Activity

An Activity defines the actions that should be performed in this service. To create a service action:

1. From the tree in the application rules side panel, choose Value Added Services > Activities. A window with a list of the service activities displays.

2. Choose the line marked with a star (*) and click into each of the fields to enter values for Activity Code, which defines the Activity, and the Description.

7.2.2 Modifying an Activity

To modify a service activity:

1. From the tree in the application rules side panel, choose Activities. A window with a list of the activities displays.
2. Choose the activity you wish to modify, and click in the Description field to change the Description.

7.2.3 Deleting an Activity

To delete an activity:

1. From the tree in the application rules side panel, choose Activities. A window with a list of the service activities displays.
2. Choose the activity you wish to delete, and 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 the section entitled "Defining Transactions" in the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*. 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 341 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 Configurator. 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 defining transactions and about this procedure, see the sections *Defining Transactions* and *Specifying a Transaction as Time-Triggered* in the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

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
<p>*These files are located in the following directory: <INSTALL_DIR>/xapidocs/api_javadocs/XSD/HTML</p>			

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	<code>createDerivedOrder()</code>

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

Criteria Parameters

The following are the criteria parameters for this transaction:

Table A–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 452 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 490 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 456](#) 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 490 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 *Selling and Fulfillment Foundation: Application Platform Configuration 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 (\leq) 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="768 786 1146 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.

A

- additional attributes, 37
 - creating, 38
 - defining a product item's, 86
 - deleting, 39
 - modifying, 38
- additional capacity
 - creating, 168
 - defining, 168
 - deleting, 170
 - modifying, 170
 - service types, 168
- Additional Capacity field, 169
- alias types, 48, 87, 156
 - creating, 48, 156
 - deleting, 50, 158
 - modifying, 49, 157
- Allow Fractional Values in Conversion field, 125
- Alternate UOM field, 72
- Always Tag Controlled field, 80
- application rules side panel, 10
- Applications Manager
 - actions, 26
 - document types, 27
 - entering dates/times, 32
 - lists, 29
 - lookup functionality, 26
 - on-line help, 32
 - special characters, 33
 - troubleshooting, 32
 - users, 29
 - layout, 8
 - starting, 7

- work area, 21
- association types, 50
 - creating, 50
 - deleting, 52
 - modifying, 51
- ATP Monitor Rule field, 78, 148
- ATP Rule field, 78, 148

B

- building
 - catalog index, 230
- business models, 2
 - marketplaces, 3
 - multi-divisional corporations, 2
 - third-party logistics models, 2
- buyers, 107

C

- Capacity Volume field, 107
- catalog
 - index building, 230
- catalogs, 130
- categories, 130
 - additional attributes
 - creating, 133
 - deleting, 134
 - modifying, 134
 - containerization, 107
- category statuses
 - creating, 131
 - deleting, 132

- modifying, 132
- classification definitions
 - create, 137
 - deleting, 138
 - modifying, 138
- classification hierarchies, 135, 143
 - classification values
 - creating, 143
 - deleting, 151
 - modifying, 145, 146
- classification purposes
 - creating, 141
 - deleting, 142
 - modifying, 142
- classifications, 135
 - classification definitions, 136
 - classification hierarchies, 143
 - classification purposes, 138
 - defining a product item's, 82
- Commodity Code field, 83
- compliance services, 179
- configuration screens
 - accessing, 11
- containers
 - defining, 105, 110
 - SKU capacity definition, 109
- Conversion Quantity to Inventory UOM field, 72
- cost posting classification, 158
- cost posting classification codes
 - deleting, 160
 - modifying, 159
- Cost Posting Classification field, 84
- Country of Origin field, 67
- Create New Item Instruction field, 88, 100, 187
- Create Work Order While Scheduling An Order
 - Requiring this Service field, 183
- Credit Without Receipt field, 68
- cross-sell item associations, 89

D

- Default Expiration Days field, 78
- Default Pricing UOM Conversion Factor field, 70
- dekitting services, 179
- delivery service items, 161

- defining for a product item, 94
- Description field, 66, 129, 173, 182

E

- ECCN Number field, 83
- environment variable
 - INSTALL_DIR, xxxiii
 - INSTALL_DIR_OLD, xxxiii
- Extended Description field, 184

F

- Fixed Capacity Units field, 163, 173
- Future Safety Factor Percentage field, 79, 149

G

- GTIN field, 66

H

- Harmonized Code field, 83
- Hazardous Materials Class field, 84
- Height field, 69, 73
- Hold Scheduling Until Associated Line Completes field, 96

I

- index
 - catalog search, 230
- inheritance
 - determining, 12
- INSTALL_DIR, xxxiii
- INSTALL_DIR_OLD, xxxiii
- Instruction URL field, 189
- Instruction Usage field, 189
- Instructions
 - defining for a product item, 98
- inventory availability safety factors, 79, 149
- inventory change services, 179
- inventory information
 - defining, 77

Inventory Is Stored In This UOM field, 124
Is Hazardous Item field, 84
Item associations
 defining a product item's, 89
item attribute overrides, 120
 creating, 120
 deleting, 122
 modifying, 122
item exclusion codes, 52, 87
 creating, 53, 55, 56, 58, 158
 deleting, 54, 57, 59
 modifying, 54, 57, 58
Item ID field, 63, 96, 163, 173
Item Instruction Code field, 88, 99, 186
item instruction codes, 39
 creating, 39
 deleting, 41
 modifying, 41
Item Instruction Type field, 88, 99, 186
item instruction types
 creating, 42
 deleting, 43
 modifying, 43
item instructions, 184
 creating, 185
 deleting, 187
 modifying, 187
item queries
 creating, 111
Item Shipment Consolidation Window _ Days
 field, 75
item statuses, 35
 creating, 36
 deleting, 37
 modifying, 37
Item Type field, 84
Item with Variations, 65
items
 as containers, 105, 110
 container attributes, 105, 110

K

Kit Code field, 66
kit components

 defining for a product item, 101
kitting services, 179

L

Lead Days field, 79, 148
Length field, 69, 73
LPN Type field, 124

M

Manufacturer's Item Description field, 67
Manufacturer's Item field, 67
Manufacturer's Name field, 67
marketplaces, 3
Master Catalog field, 63, 65
Master Catalog ID field, 129
master catalogs, 63
 creating, 128
 deleting, 130
 modifying, 129
master unit of measure
 modifying for delivery service items, 165
master units of measure
 creating for delivery service items, 165
 creating for provided service items, 175
 deleting for delivery service items, 166
 deleting from provided service items, 176
 modifying for provided service items, 175
Maximum Order Quantity field, 67
Maximum weight that can be placed in the container
 field, 107
Minimum Order Quantity field, 67
multi-divisional corporations, 2

N

NAICS Code field, 83
Negative Package Tolerance Qty field, 72
NMFC Class field, 83
NMFC Code field, 83
node attribute overrides
 defining a product item's, 94
Node needs to be notified at least ___ hours prior to

expected time of shipment field, 75
Not Tag Controlled field, 80
Number of Secondary Serials, 80

O

Onhand Safety Factor Percentage field, 79, 149
Onhand Safety Factor Quantity field, 79, 149
Operational Configuration Complete field, 85
ordering units of measure, 71
Ordering UOM field, 70
Orders Can Be Placed Using This UOM field, 73, 124
organization levels, 12
 rules, 15
organization rules, 15
 loading another organization's rules, 19
 overriding, 16

P

Picking Type field, 84
Positive Package Tolerance Qty field, 72
Pricing organization, 12, 15
pricing units of measure
 creating, 126
 deleting, 127
 modifying, 127
Pricing UOM Different Is From Inventory UOM, 70
Pricing UOM field, 70
Product field, 105
Product items
 defining additional attributes, 86
product items, 61
 creating, 62
 defining associated service items, 94
 defining classifications, 82
 defining instructions, 98
 defining inventory information, 76
 defining item associations, 89
 defining kit components, 101
 defining node attribute overrides, 94
 defining primary information, 64
 defining units of measure, 68
 adding an alternate unit of measure, 71

 deleting an alternate unit of measure, 73
 modifying an alternate unit of measure, 73
 deleting, 111
 modifying, 63
 tag controlled, 76
Product Line field, 84
Product Quantity field, 96
provided service items, 171
 defining for a product item, 94

Q

Quantity field, 70, 108
quantity units of measure
 creating, 123
 deleting, 126
 modifying, 125

R

Region Schema Used To Apply The Additional Capacity field, 169
Reserved for Items having the following
 Containerization Categories field, 107
Reserved for the following Buyers field, 107
restricting containers, 107
Return Window field, 68
Returnable field, 68
Run Quantity field, 184

S

Schedule B Code field, 83
Sequence No field, 88, 99, 186, 189
Serial Numbers Tracked In Inventory field, 79
Serial Numbers Tracked In Returns And Outbound field, 80
service activities, 179
 creating, 188
 details, 188
service activities details
 adding, 188
 deleting, 190
 modifying, 190

Service Complexity Level field, 167
service complexity levels
 creating, 46
 defining, 46
 deleting, 48
 modifying, 47
Service Item Capacity Quantity Per field, 96
Service Item ID For Serviced Area Search field, 167
Service Item Pricing Quantity Per field, 96
Service Item UOM For Service Area Search field, 167
service options, 172
 creating for a provided service item, 172
 creating for delivery service items, 162
 defining for delivery service items, 162
 deleting for delivery service items, 164
 deleting from a provided service item, 174
 modifying for a provided service item, 174
 modifying for delivery service items, 164
service skills
 creating, 44
 defining, 43
 deleting, 46
 modifying, 45
Service Time Offset field, 96
Service tool field, 67
Service Type field, 169
Service Type ID field, 167
service types, 46
 additional capacity, 168
 defining, 166, 176
 delivery
 creating, 166
 defining, 166
 deleting, 168
 modifying, 168
 provided
 creating, 176
 deleting, 178
 modifying, 178
Short Description field, 63, 65, 129, 163, 173, 182
SKU capacity definition
 deleting, 109
SKU Unit Size Code field, 108

Sometimes Tag Controlled field, 80
Status field, 66, 182
Storage Type field, 84
supersession on item associations, 90

T

Tax Product Code field, 84
Taxable field, 67
Text field, 189
third-party logistics models, 2
Time Sensitive field, 78
Track FIFO in Inventory, 79, 148

U

UCC Packaging Indicator field, 124
UNSPSC field, 83
UOM Code field, 124
UOM Conversion Precision field, 125
UOM Description field, 124
UOM field, 63, 70, 96, 163, 173
UOM (Units Of Measure)
 pricing, 126
 quantity, 123
up-sell item associations, 90
Use Item Instruction Code field, 88, 99, 186
Used as a Container for Packing field, 63

V

value added services. See VAS
VAS (Value Added Services), 179
 creating, 181
 defining, 180
 deleting, 191
 modifying, 191
Velocity Code field, 85
Volume Allowance field, 107

W

Weight field, 69, 73
Width field, 69, 73

work orders, 179