

Sterling Commerce

An IBM Company

Selling and Fulfillment Foundation: Extending the Database Guide

Release 8.5

Last updated in HF19

July 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 ehcache 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

1	Checklist for Customization Projects	
1.1	Customization Projects.....	3
2	Guidelines for Extending Databases	
2.1	About Extending the Database	3
2.1.1	Entity Relationship Diagrams	3
2.1.2	Entity Database XML Files	4
2.2	Guidelines for Adding Columns to a Standard Table	4
2.3	Guidelines for Adding Support for Data Compression for Columns	5
2.3.1	Data Compression Logic.....	6
2.4	Guidelines for Adding Non-Unique Indices to a Standard Table	6
2.5	Guidelines for Adding Foreign Key Elements to a Standard Table	7
2.6	Guidelines for Adding Text Search Index Elements to a Standard Table	7
2.7	Coding Guidelines To Avoid Deadlocks	8
3	Extending Database Tables	
3.1	Adding a Column to a Standard Table	11
3.1.1	Features Requiring Multiple Extensions When Adding a Column to a Standard Table	15
3.2	Extending a Catalog Search	16
3.3	Adding Unique Tag Identifiers and Descriptors to a Standard Table	21
3.3.1	Extending Tables When Adding Unique Tag Identifiers.....	22
3.3.2	Extending Tables When Adding Unique Tag Descriptors	23
3.4	Adding Non-Unique Indices to a Standard Table	24
3.5	Adding Foreign Key Elements to a Standard Table	25
3.6	Adding Text Search Indices to a Standard Table	28
3.7	Enabling Case Insensitive Searches	29
3.7.1	Modifying an Entity XML.....	31
3.8	Creating Custom and Hang-off Tables	32
3.8.1	Steps to create a custom table.....	34

3.8.2	Steps to create a hang-off table	40
3.8.3	Purging Data from Hang-Off Tables.....	47
4	Extending the Data Types Files	
4.1	Extending the Data Types Files	49
4.1.1	Extending the Data Type Map File.....	49
4.1.2	Extending the Data Type File	49
5	Generating Audit References for Entities	
5.1	About Extending Audit References	51
6	Extending API Templates	
6.1	About Extending API Templates.....	55
6.2	Including Extended Attributes in the API Template.....	56
6.3	Including Custom and Hang-Off Entities in the API Template	57
6.4	Configuring Services for Custom and Hang-off APIs	59

Index

Checklist for Customization Projects

This chapter provides a high-level checklist for the tasks involved in customizing or extending an Application.

1.1 Customization Projects

Projects to customize or extend an Application vary with the type of changes that are needed. However, most projects involve an interconnected series of changes that are best carried out in a particular order. The checklist identifies the most common order of customization tasks and indicates which guide in the documentation set provides details about each stage.

1. Prepare your development environment

Set up a development environment that mirrors your production environment, including whether you deploy Application on a WebLogic, WebSphere, or JBoss application server. Doing so ensure that you can test your extensions in a real-time environment.

You install and deploy Application in your development environment following the same steps that you used to install and deploy Application in your production environment. Refer to Application system requirements and installation documentation for details.

An option is to customize Application with Microsoft COM+. Using COM+ provides you with advantages such as increased security, better performance, increased manageability of server applications, and support for clients of mixed environments. If this is your choice, see the *Platform Customization Basics Guide* about additional installation instructions.

2. Plan your customizations

Draft

Are you adding a new menu entry, customizing the Sign In screen and logo, creating new themes, customizing views and wizards, or adding new screens? Each type of customization varies in scope and complexity. For background, see the *Platform Customization Basics Guide*, which summarizes the types of changes that you can make.

Important guidelines about file names, keywords, and other conventions are found in the *Platform Customization Basics Guide*.

3. Extend the Database

For many customization projects, the first task is to extend the database so that it supports the other UI or API changes that you make later. For instructions, see the *Platform Extending the Database Guide* which include information about the following topics:

- Important guidelines about what you can and cannot change in the database.
- Information about modifying APIs. If you modify database tables so that any APIs are impacted, you must extend the templates of those APIs or you cannot store or retrieve data from the database. This step is required if table modifications impact an API.
- How to generate audit references so that you improve record management by tracking records at the entity level. This step is optional.

4. Make other changes to APIs

Application can call or invoke standard APIs or custom APIs. For background about APIs and the services architecture in Application, including service types, behavior, and security, see the *Platform Customizing APIs Guide*. This guide includes information about the following types of changes:

- How to invoke standard APIs for displaying data in the UI and also how to save the changes made to the UI in the database.
- Invoke customized APIs for executing your custom logic in the extended service definitions and pipeline configurations.
- APIs use input and output XML to store and retrieve data from the database. If you don't extend these API input and output XML

Draft

files, you may not get the results you want in the UI when your business logic is executing.

- Every API input and output XML file has a DTD and XSD associated to it. Whenever you modify input and output XML, you must generate the corresponding DTD and XSD to ensure data integrity. If you don't generate the DTD and XSD for extended Application XMLs, you may get inconsistent data.

5. Customize the UI

Sterling Commerce applications support several UI frameworks. Depending on your application and the customizations you want to make, you may work in only one or in several of these frameworks. Each framework has its own process for customizing components like menu items, logos, themes, and etc. Depending on the framework you want, consult one of the following guides:

- *Platform Customizing Console JSP Interface Guide*
- *Platform Customizing the Swing Interface Guide*
- *Platform Customizing User Interfaces for Mobile Devices Guide*
- *Platform Customizing the Rich Client Platform Guide* and *Platform Extensibility Tool for the Rich Client Platform Guide*
- *Customizing the Web UI Framework Guide*

6. Extend Transactions

You can extend the standard Application to enhance the functionality of your implementation of Application and to integrate with external systems. For background about transaction types, security, dynamic variables, and extending the Condition Builder, see the *Platform Extending Transactions Guide* and *Platform Extending the Condition Builder Guide*. These guides includes information about the following types of changes:

- How to extend Application Condition Builder to define complex and dynamic conditions for executing your custom business logic and using a static set of attributes.
- How to define variables to dynamically configure properties belonging to actions, agents, and services configurations.

Draft

- How to set up transactional data security for controlling who has access to what data, how much they can see, and what they can do with it.
 - How to create custom time-triggered transactions. You can invoke and schedule these custom time-triggered transactions in much the same manner as you invoke and schedule Application standard time-triggered transactions. Finally, you can coordinate your custom, time-triggered transactions with external transactions and run them either by raising an event, calling a user exit, or invoking a custom API or service.
7. Build and deploy your customizations or extensions

After performing the customizations that you want, you must build and deploy your customizations or extensions. First, build and deploy these customizations or extensions in the test environment for verification. When you are ready, repeat the same process to build and deploy your customizations and extensions in the production environment. For instructions, see the *Platform Customization Basics Guide* which includes information about the following topics:

- How to build and deploy standard resources, database, and other extensions (such as templates, user exits, java interfaces).
- How to build and deploy Enterprise-Level extensions.

Draft

Guidelines for Extending Databases

2.1 About Extending the Database

Certain aspects of the Selling and Fulfillment Foundation database cannot be modified. If you try to make these modifications, your data is not harmed, but your attempted changes are not incorporated into the Selling and Fulfillment Foundation database. Selling and Fulfillment Foundation does not permit modification of the following:

- Existing columns of tables
- Primary keys of tables
- Unique keys of tables
- Views

When planning extensions to the database, consider the implications of your changes and how they may impact other areas.

Important: If you modify a table and your deployment uses Sterling Analytics , the view associated with the table must also be modified.

2.1.1 Entity Relationship Diagrams

To learn more about the Selling and Fulfillment Foundation database, see the entity relationship diagrams (ERDs) using the <INSTALL_DIR>/xapidocs/erd/html/erd.html file. These ERDs provide the following information:

- Indicate which tables can be extended by adding columns.
- Indicate which tables can have hang-off relationship.

- Relationships between tables (to help you understand the relationship between logical entities such as orders, shipments, and payments).
- Indices details. Each table is indexed by a primary key. Most tables also have a unique index that is constituted of the columns that make the logical unique key. In addition, some tables have alternate indices to support queries.
- Views that indicate how several tables interact.

2.1.2 Entity Database XML Files

The standard tables that are shipped with Selling and Fulfillment Foundation are defined in a set of entity XML files, also known as *database definition* XML files. Each entity XML file may contain several table definitions. To learn more about these tables, see the files in the `<INSTALL_DIR>/repository/entity` directory. Within these entity XML files, an entity represents a table and an attribute represents a column.

The following sections describe the general guidelines to follow when adding columns, indices and foreign key elements.

2.2 Guidelines for Adding Columns to a Standard Table

When extending the columns of standard tables, keep the following considerations in mind:

- You can only add columns to tables as specified in the ERDs.
- You cannot remove or modify any columns.
- You can add columns either before or after the installation of Selling and Fulfillment Foundation.
- For all columns added to a Selling and Fulfillment Foundation table, you must provide a default value that is relevant to the database framework.
- You cannot use nullable columns for the following fields:
 - Primary Key Attributes
 - Entity Relationships

Hence, in the entity XML, `Nullable="true"` is allowed for all columns except the ones noted above.

- You cannot add columns with a data type of `Long`.
- When using components of Selling and Fulfillment Foundation (such as events and user exits) that read in a map or publish a map (such as the `GetOrderNoUE` user exit), extended fields in the maps are prefixed with `Extn_`.

Note: In Oracle database, data type CLOB is generated instead of LONG for new installations. However, for upgrades, existing LONG columns will remain unchanged.

2.3 Guidelines for Adding Support for Data Compression for Columns

You can enable data compression for new and existing columns of Standard, Custom, or Hang-off tables. The data compression allows a reduction in size for tables when being implemented.

- For newly added columns, you can enable data compression by adding both `CompressionSupported="true"` and `UseCompression="true"` in the entity XML.
- For existing columns in a standard table, you can enable data compression by overriding the value of the `UseCompression` attribute.

Note: you may only enable data compression for existing columns that have the `CompressionSupported` attribute set to `True`.

When adding support for data compression to the columns of standard tables, keep the following considerations in mind:

- Data Compression should only be used for columns containing a large amount of text data that is typically not queried directly, such as audit records, error stack traces, and XML template data. The nature of these records can consume a significant amount of space within the database.

- Columns using `CompressionSupported="true"` do not support query operations from list APIs. This can be overridden by setting the `QueryAllowed` attribute to `True` in the entity XML.
- By default, columns using `CompressionSupported="true"` cannot be used as part of an index or unique constraint. This can be overridden by setting the `AllowCompressedColumns` attribute to `True` on the index element in the entity XML.
- Primary Key columns cannot be marked as compressible columns.
- Data compression should be enabled only if the maximum size of the column is ≥ 500 bytes. Any column marked for data compression with less than 500 bytes results in warnings.
- By default, the data compression is done using the GZip algorithm. You can override this default data compression logic by setting the `yfs.db.compression.class=<class name>` property in the `customer_overrides.properties` file. In the `<class name>`, specify the name of your custom class which contains the data compression logic. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

2.3.1 Data Compression Logic

The data for compressible columns is compressed using the GZip algorithm. You can override this default data compression logic and provide your custom compression logic by implementing the `SCICustomDataCompressor` interface.

You must enter the name of the custom class that implements the `SCICustomDataCompressor` interface in the `yfs.db.compression.class=<class name>` property in the `customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

2.4 Guidelines for Adding Non-Unique Indices to a Standard Table

When adding non-unique indices, use a naming convention that differs from the Selling and Fulfillment Foundation convention, which is

`<tablename>_i<1+n>`. Using your own naming convention prevents your indices from accidentally being dropped during upgrades. The following considerations are also recommended:

- Adding a prefix that doesn't start with `y`.
- Prefix your non-unique indices with `EXTN_` for easier identification.
- Unique indices are not allowed for tables.
- Column names for indices must be valid.
- Index names should not exceed 18 characters.

2.5 Guidelines for Adding Foreign Key Elements to a Standard Table

Currently foreign key relationships in extended columns of tables are restricted to only the `YFS_PERSON_INFO` table. When exposing foreign key elements, the following validations are performed:

- The parent table name must be `YFS_PERSON_INFO`.
- The parent column name must be a primary key of the `YFS_PERSON_INFO` table.

2.6 Guidelines for Adding Text Search Index Elements to a Standard Table

When adding text search index elements to standard tables, keep the following considerations in mind:

- You can add the text search element only to non-transactional tables.
- The searches performed for the text search indices are case insensitive.
- You can define more than one text search indices per entity.
- You cannot define multiple columns on a text index.
- Sterling Commerce supports `CTXCAT` and `CONTEXT` text search index types on Oracle.
- Sterling Commerce recommends that you define both the extended entity and the text search index in the same extension XML file.

2.7 Coding Guidelines To Avoid Deadlocks

Deadlock refers to a specific condition in a database, when two processes are waiting for the other process to release a resource. For example when one client application holds a lock on a table and attempts to obtain the lock on a second table that is held by another client application, this may lead to a deadlock if the other application attempts to obtain the lock that is held by the first application.

To circumvent the deadlock problem, Sterling Commerce recommends that you need to sort the information to be accessed in a certain order before grabbing locks. This is applicable to situations where you need to grab multiple inventory item locks within a single transaction boundary. However, you do not need to sort if you call the APIs to process single items per transaction commit.

Reading Uncommitted data in DB2 Database

In DB2, when you select a record from a table, a read lock is obtained on the record. If the record being selected has been updated but not committed, the thread waits until it commits the changes. Alternatively you could read the record with Uncommitted Read (UR) in which case the latest value that has been updated is provided to the user.

You can read uncommitted data from any list API by enabling the `ReadUncommitted` attribute to `Y` in its input XML. To achieve this, you must customize the individual JSP and pass the `ReadUncommitted` attribute as a hidden attribute. For example:

```
<input type="hidden" name="xml:/Order/@ReadUnCommitted" value="Y"/>
```

As a result the locking scenario is circumvented in DB2 database. Locking is the default in DB2.

It is not mandatory to pass this flag. However, if you set this flag to `Y`, the system is forced to read uncommitted data. For example, a transaction, `T1`, updates Table `TAB-1` but the transaction's data is not committed. If the `ReadUncommitted` flag is set to `Y`, other transactions can read the uncommitted data in Table `TAB-1`.

Before setting this flag, evaluate concurrent transactions to determine whether a situation exists in which a deadlock is occurring. If no such situation occurs, the flag should remain at its default setting.

This behavior is different from Oracle, hence if you are writing custom code on DB2 you should understand this behavior to avoid lock escalations.

Extending Database Tables

3.1 Adding a Column to a Standard Table

You add columns to tables by modifying the entity database extension XML files and then rebuilding the Selling and Fulfillment Foundation database and JAR files. After Selling and Fulfillment Foundation has been rebuilt, the APIs recognize these added columns and use them when storing and retrieving data.

To add a column to a standard table:

1. Copy the `<INSTALL_DIR>/repository/entity/extensions/Extensions.xml.sample` file as `<INSTALL_DIR>/extensions/global/entities/<your_filename>.xml` file OR modify your existing extension XML file.
2. Edit the `<your_filename>.xml` file to add a new entity tag as shown in [Example 3-1](#) for each table you want to extend. If the tag already exists, use the existing one. For a description of the XML attributes, see [Table 3-1](#) on page 3-12.

Example 3–1 Sample XML for Extending Columns

```

<!-- element exposed to create a column -->
<DBSchema>
  <Entities>
    <Entity TableName="REQUIRED">
      <Attributes>
        <Attribute ColumnName="REQUIRED" DataType="" DecimalDigits=""
          CompressionSupported="false" UseCompression="false"
          QueryAllowed="false" DefaultValue="" Description="" Nullable="false"
          Size="1" Type="REQUIRED" XMLName="" XMLGroup="" SqlServerDataType=""
        />
      </Attributes>
    </Entity>
  </Entities>
</DBSchema>

```

Table 3–1 Attributes in the XML <Attributes> Tag

Attribute	Description
ColumnName	Required. Name of the column added to this table. The ColumnName must start with EXTN_.
DataType	Optional. Valid values are available in the <INSTALL_DIR>/repository/datatypes/datatypes.xml file.
DecimalDigits	Optional. Number of digits of precision required after the decimal. Needed only for numeric fields.
CompressionSupported	Optional. Attribute used to indicate whether or not the data compression is supported for this column. Valid values are True or False. If True compression support is enabled. Note: If the data inserted into a column has both CompressionSupported and UseCompression attributes set to True, then this attribute should not be set back to False. Doing so would result in all compressed data being retrieved without any decompression.
UseCompression	Optional. Attribute used to compress data for this column. Valid values are True or False. If True data is compressed. Note: The value of this attribute should be set to True only if the CompressionSupported attribute is set to True.

Table 3–1 *Attributes in the XML <Attributes> Tag*

Attribute	Description
QueryAllowed	Optional. Attribute used to enable a compressible column to be used for queries in a List API. Valid values are True or False. If True the column can be used for queries in the List APIs. Note: If the value of the CompressionSupported attribute is set to true then the value of this attribute should also be set to True.
DefaultValue	Required. Used as is for the defaults clause in your database.
Description	Optional. Description of column usage.
Nullable	Optional. Attribute used to describe the nullable value of a field. Default is false. Nullable=true is allowed for all columns except Primary Key Attributes and Entity Relationships.
Size	Size of the database column.
Type	Required. Data type of the database column. This attribute also determines the type of attribute in the Java classes that are generated and the format of the attribute in the XML. The valid types are CHAR, VARCHAR2, NUMBER, DATE, and TIMESTAMP. If you are using Microsoft SQL Server [®] and want to specify a data type as TEXT in the database, you also need to use the SqlServerDataType attribute and specify the attribute value as TEXT. Note: If DATE is specified, only the calendar date is stored. If TIMESTAMP is specified, the calendar date and time are stored.
XMLName	XML name of the attribute, if it is different from the name of the attribute. Choose a name that does not conflict with the base extension. It is recommended that you use Extn as a prefix. It is also strongly recommended that you use the same convention for arriving at the XMLName as the Selling and Fulfillment Foundation base product does: Make each letter following the underscore in the column name upper case, and the rest lower case. Then, remove the underscores. Thus, Extn_Item_Id should be: ExtnItemId.

Table 3–1 Attributes in the XML <Attributes> Tag

Attribute	Description
XMLGroup	If present, indicates the child tag in which the attribute is present. If the attribute is not present in the XML, use the NOT_SHOWN string. The XMLGroup must be Extn. Thus, the data for the extended columns is in a separate element in the API XML output.
VirtualDefaultValue	Optional. This attribute is applicable to Null Columns. When a null is returned from the database, it is stored in memory as the virtual default value.
SqlServerDataType	Optional. Pertains only to Microsoft SQL Server databases. If you see a warning about the row size being too long, specify one or more of your larger columns as "TEXT". Columns of type TEXT are not included in the maximum row size calculation for a table.
ForceUpperCase	Optional. If a "case insensitive" search is required for a text column, this field should be set to "True". If set to "True", the system converts the data entered in this field to uppercase.

3. Create a new `Attribute` tag for each column you want to add to the table.
4. Use the database verification tool `dbverify` for generating scripts to add the columns to your database.

Note: On Microsoft SQL Server, the total length of all extended columns should not exceed 900 bytes. If Microsoft SQL Server throws a warning that the row size exceeds the maximum length, change the data type of one or more of your columns to TEXT and then specify TEXT for the `SqlServerDataType` attribute as described in [Table 3–1](#).

5. Extend the corresponding API templates by following the steps described in [Section 6.1, "About Extending API Templates"](#).
6. Build and Deploy your extensions.

A special case of extending columns for adding unique tag identifiers or descriptors is explained in [Section 3.3, "Adding Unique Tag Identifiers and Descriptors to a Standard Table"](#).

3.1.1 Features Requiring Multiple Extensions When Adding a Column to a Standard Table

In addition to extending standard tables, you can extend the following features using the specified guidelines:

- **Classification Inheritance**—If you are adding columns to extend attributes in the YFS_ITEM table and you want to make these attributes available for classification inheritance, a duplicate entity tag must be added to your XML for the YFS_CLASS_ITEM_ATTR table. Additionally, `Nullable` should be set to "true" and `DefaultValue` should not be passed for these attributes. For more information on defining item attributes at the classification level, refer to the *Catalog Management: Configuration Guide*.
- **Item Entitlements**—If you are adding columns to extend attributes in the YFS_CUSTOMER table and you want to make these attributes available for entitlement rule assignment, a duplicate entity tag must be added to your XML for the YFS_ENTITLE_RULE_ASSIGNMENT table. Additionally, `Nullable` should be set to "true" and `DefaultValue` should not be passed for these attributes. For more information on entitlement rule assignment, refer to the *Business Center: Item Administration Guide*.
- **Pricelist Assignments**—If you are adding columns to extend attributes in the YFS_CUSTOMER table and you want to make these attributes available for pricelist assignment, a duplicate entity tag must be added to your XML for the YFS_PRICELIST_ASSIGNMENT table. Additionally, `Nullable` should be set to "true" and `DefaultValue` should not be passed for these attributes. For more information on pricelist assignment, refer to the *Business Center: Pricing Administration Guide*.
- **Pricing Rule Assignments**—If you are adding columns to extend attributes in the YFS_CUSTOMER table and you want to make these attributes available for pricing rule assignment, a duplicate entity tag must be added to your XML for the YFS_PRICING_RULE_ASSIGNMENT table. Additionally, `Nullable` should be set to "true" and `DefaultValue` should not be passed for these attributes. For

more information on pricing rule assignment, refer to the *Business Center: Pricing Administration Guide*.

3.2 Extending a Catalog Search

You can extend a catalog search to provide broader search capabilities by updating information in the catalog search index file. To update the index file, modify the corresponding extended XML configuration file. The following components of catalog search are extensible:

- Search system configurations
- Locales and the corresponding analyzer
- Query parser
- Attributes
- Asset types

To extend a catalog search:

1. Copy the `<INSTALL_
DIR>/repository/xapi/template/merged/resource/extn/ExtnCatalogSearchConfigProperties.xml` sample file as `<INSTALL_
DIR>/extensions/global/template/resource/extn/ExtnCatalogSearchConfigProperties.xml` file OR modify your existing extension XML file. [Example 3–2](#) shows a sample `ExtnCatalogSearchConfigProperties.xml` file.

Example 3–2 Sample XML for ExtnCatalogSearchConfigProperties.xml

```

<SearchConfigurations>
  <SearchSystemConfigurations>
    <MergeFactor Value="2"/>
    <MaxMergeDocs Value="2147483647"/>
  </SearchSystemConfigurations>
  <IndexSets>
    <IndexSet Name="CatalogIndex">
      <Locales>
        <Locale LocaleCode="en_US"
          SynonymFile="/properties/EnglishSynonym.properties"/>
        <Locale LocaleCode="fr_FR"
          QueryParserClass="package.FrenchQueryParser.class"
          AnalyzerClass="package.FrenchAnalyzer.class"
          SynonymFile="/properties/FrenchSynonym.properties" />
      </Locales>
      <Entities>
        <Entity Name="Item">
          <Attributes>
            <Attribute XMLName="MyExtendedDescription" DefaultWeight="1.0"
              Index="ANALYZED" Store="Y" UseSynonyms="N" Searchable="Y"
              Sortable="N"/>
          </Attributes>
          <Entity Name="CategoryItem" RelationshipName="Category_ItemList">
            <Entity Name="Category" RelationshipName="Category">
              <Attributes>
                <Attribute XMLName="MyCategoryExtendedDescription"
                  DefaultWeight="1.0" Index="ANALYZED" Store="Y" UseSynonyms="N"
                  Searchable="Y" Sortable="N"/>
              </Attributes>
            </Entity>
          </Entity>
          <Entity Name="Asset" RelationshipName="AssetList">
            <Attributes>
              <Attribute Type="MyManual" DefaultWeight="1.0" Index="ANALYZED"
                Store="N" UseSynonyms="N" Searchable="Y" Sortable="N"/>
            </Attributes>
          </Entity>
        </Entity>
      </Entities>
    </IndexSet>
  </IndexSets>
</SearchConfigurations>

```

- To modify system configuration information for the index file, modify the elements in the `SearchSystemConfigurations` section of `ExtnCatalogSearchConfigProperties.xml`. [Table 3–2](#) describes the `SearchSystemConfiguration` elements in `ExtnCatalogSearchConfigProperties.xml`.

Table 3–2 SearchSystemConfiguration Elements in ExtnCatalogSearchConfigProperties.xml

Element	Description
MergeFactor Value	Specifies the number of documents that the Apache Lucene open source search engine stores in memory before writing them to disk as a single segment. For more information about setting this value, see: http://lucene.apache.org
MaxMergeDocs Value	Specifies the number of documents that can be contained in one segment. For more information about setting this value, see: http://lucene.apache.org

- To add or modify locale information for the index file, add or modify the elements in the `Locales` section of `ExtnCatalogSearchConfigProperties.xml`. [Example 3–2](#) shows a sample `ExtnCatalogSearchConfigProperties.xml` file in which a synonym file has been added to the US-English locale, and a French locale has also been added. [Table 3–3](#) describes the `Locale` elements in `ExtnCatalogSearchConfigProperties.xml`.

Table 3–3 Locale Elements in ExtnCatalogSearchConfigProperties.xml

Element	Description
LocaleCode	Specifies the locale code for the index file. If you are adding a locale, ensure that you specify a corresponding analyzer. US-English is the default locale.
QueryParserClass	Specifies the path to the query parser. Selling and Fulfillment Foundation provides the Lucene query parser by default.

Table 3–3 *Locale Elements in ExtnCatalogSearchConfigProperties.xml*

Element	Description
AnalyzerClass	Specifies the path to the analyzer. Selling and Fulfillment Foundation provides corresponding analyzers for the default locale.
SynonymFile	Specifies the path to the synonym file for the corresponding locale. Use the synonym file to configure related terms for keyword searches. For more information about configuring synonyms, see the <i>Selling and Fulfillment Foundation: Localization Guide</i> .

- To add item information for the index file, add the attributes in the Item section of `ExtnCatalogSearchConfigProperties.xml`. [Example 3–2](#) shows a sample `ExtnCatalogSearchConfigProperties.xml` file in which the `MyExtendedDescription` Item attribute has been added. [Table 3–4](#) describes Item attribute elements in `ExtnCatalogSearchConfigProperties.xml`.

Note: You can add attributes to the extended XML configuration file for the catalog search index, but you cannot modify attributes nor add entities. See `<INSTALL_DIR>/repository/xapi/template/merged/resource/CatalogSearchConfigProperties.xml` for a list of attributes that are supported by default.

- To add catalog information pertaining to the index file, add the attributes in the Category section of `ExtnCatalogSearchConfigProperties.xml`. [Example 3–2](#) shows a sample `ExtnCatalogSearchConfigProperties.xml` file in which the `MyCategoryExtendedDescription` Category attribute has been added. [Table 3–4](#) describes the Category attribute elements in `ExtnCatalogSearchConfigProperties.xml`.
- To add asset information for the index file, add attributes in the Asset section of `ExtnCatalogSearchConfigProperties.xml`. [Example 3–2](#) shows a sample `ExtnCatalogSearchConfigProperties.xml` file in which the asset with the asset type `MyManual` added. [Table 3–4](#)

describes Asset attribute elements in `ExtnCatalogSearchConfigProperties.xml`.

Note: Selling and Fulfillment Foundation calls the `YCMParseAssetUE` user exit to parse the content of the asset for the corresponding asset type.

Table 3–4 Item, Category, and Asset Attribute Elements in `ExtnCatalogSearchConfigProperties.xml`

Element	Description
Attribute XMLName -or- Attribute Type	For item and category attributes, specifies the XML attribute name for the field. For asset attributes, specifies the asset type in the database.
Default Weight	Specifies the weight given to the term. The default value is 1.
Index	Specifies one of the following options for storing field values: Analyzed - stores the value as searchable segments. For example, if a user searches for Desktop Computers and the index parameter for the field is analyzed, the search results include items with the terms Desktop Computer, Desktop, and Computer. Non_analyzed - stores the value as it appears in the database, requiring an exact match to return the value. In the previous example, if the search for Desktop Computers was performed on a non_analyzed field, the search would return only items with the term Desktop Computer.
Store	Specifies one of the following options for storing field values: Y - stores the value of the attribute in the index so that it may be returned as search output. N - Does not store the value and only keeps data in the index for the attribute that is in a proprietary format and used only during the search.

Table 3–4 Item, Category, and Asset Attribute Elements in ExtnCatalogSearchConfigProperties.xml

Element	Description
UseSynonyms	Specifies one of the following options for including synonyms in the search: Y - indicates that search queries include synonyms. N - indicates that search queries do not include synonyms.
Searchable	Specifies one of the following search options for the field: Y - indicates that the field is included as possible search criteria when creating a search query. N - indicates that the field is not included as possible search criteria when creating a search query. For example, the Searchable parameter for the Is_Superseded field in the default XML file is set to N and the Store parameter is set to Y. Users cannot search on the Is_Superseded field in a query. However, queries of superseded items that are obsolete return the superseding items.
Sortable	Specifies one of the following sort options for the field: Y - indicates that search results are sorted by this field. N - indicates that search results are not sorted by this item.

3.3 Adding Unique Tag Identifiers and Descriptors to a Standard Table

The Selling and Fulfillment Foundation default tag identifiers are Batch Number, Revision Number, and Lot Number. You may have a need to extend the Selling and Fulfillment Foundation Database to define unique tag identifiers or descriptors.

Sterling Commerce recommends that the data type of any unique tag identifiers or descriptors that you add be CHAR or VARCHAR.

Note: Whenever you extend the tag attributes you must also extend the console because the templates for the APIs do not contain these extended tag attributes.

For example, if you work in the metal industry, you may want to use a custom tag identifier named Steel which has both Mill and Grade attributes. Since these are not supplied by default in Selling and Fulfillment Foundation, you must extend the set of tables listed below to include the Steel tag identifier column in each table.

3.3.1 Extending Tables When Adding Unique Tag Identifiers

You must extend each of the following tables whenever you add unique tag identifiers to the Selling and Fulfillment Foundation database:

- YFS_COUNT_RESULT_TAG
- YFS_COUNT_TAG
- YFS_INVENTORY_AUDIT
- YFS_INVENTORY_TAG
- YFS_ITEM_TAG - The data type to be used for this table is CHAR(2).
- YFS_MOVE_REQUEST_LINE_TAG
- YFS_ORDER_KIT_LINE_SCHEDULE
- YFS_ORDER_KIT_LINE_SCHEDULE_H
- YFS_ORDER_LINE_REQ_TAG
- YFS_ORDER_LINE_REQ_TAG_H
- YFS_ORDER_LINE_SCHEDULE
- YFS_ORDER_LINE_SCHEDULE_H
- YFS_ORDER_LINE_RESERVATION
- YFS_RECEIPT_LINE
- YFS_RECEIPT_LINE_H
- YFS_SHIPMENT_LINE_REQ_TAG

- YFS_SHIPMENT_LINE_REQ_TAG_H
- YFS_SHIPMENT_TAG_SERIAL
- YFS_SHIPMENT_TAG_SERIAL_H
- YFS_WORK_ORDER_COMP_TAG
- YFS_WORK_ORDER_COMP_TAG_H
- YFS_WORK_ORDER_TAG
- YFS_WORK_ORDER_TAG_H

3.3.2 Extending Tables When Adding Unique Tag Descriptors

You must extend each of the following tables whenever you add unique tag descriptors to the Selling and Fulfillment Foundation database:

- YFS_COUNT_RESULT_TAG
- YFS_COUNT_TAG
- YFS_INVENTORY_TAG
- YFS_ITEM_TAG The data type to be used for this table is CHAR(2).
- YFS_ORDER_LINE_REQ_TAG
- YFS_ORDER_LINE_REQ_TAG_H
- YFS_RECEIPT_LINE
- YFS_RECEIPT_LINE_H
- YFS_SHIPMENT_LINE_REQ_TAG
- YFS_SHIPMENT_LINE_REQ_TAG_H
- YFS_SHIPMENT_TAG_SERIAL
- YFS_SHIPMENT_TAG_SERIAL_H
- YFS_WORK_ORDER_COMP_TAG
- YFS_WORK_ORDER_TAG
- YFS_WORK_ORDER_TAG_H

3.4 Adding Non-Unique Indices to a Standard Table

You can add non-unique indices to entities. You add indices to a standard Selling and Fulfillment Foundation table, by adding an `Index` element in the extension XML for that table.

To add non-unique indices to a standard table:

1. Copy the `<INSTALL_DIR>/installed_data/repository/entity/extensions/Extensions.xml.sample` file as `<INSTALL_DIR>/extensions/global/entities/<your_filename>.xml` file OR modify your existing extension XML file.
2. Edit the `<your_filename>.xml` file to add non-unique indices as shown in [Example 3–3](#) for each table you want to extend. For a description of the XML attributes, see [Table 3–5](#) on page 3-25.

Example 3–3 Sample XML for Adding Non-Unique Indices

```
<!-- element exposed to create index -->
<DBSchema>
<Entities>
  <Entity TableName="REQUIRED">
    .
    .
    <Indices>
      <Index Name="REQUIRED" AllowCompressedColumns="false">
        <Column Name="REQUIRED" />
        .
        .
      </Index>
      .
    </Indices>
    .
  </Entity>
</Entities>
</DBSchema>
```

Table 3–5 Creating Non-Unique Indices on a Selling and Fulfillment Foundation Table

Attribute	Description
Entity	
TableName	Required. Name of the table for which the indices are added, For example: YFS_ITEM.
Entity/Index	
Name	Required. The name of the custom index. Name should start with a prefix EXTN_
AllowCompressedColumns	Optional. If True, the index is allowed to contain columns marked with CompressionSupported attribute set to True.
Entity/Index/Column	
Name	Required. The name of the column for which the index is added. Create a new <Column Name/> for each column for which the index is added.

3. Create a new `Index` tag for each index you want to add to the column.
4. Extend the corresponding API templates to include the non-unique indices by following the instructions in [Chapter 6, "Extending API Templates"](#).
5. Build and Deploy your extensions.

3.5 Adding Foreign Key Elements to a Standard Table

A foreign key relationship is a relationship between an extended column in any Selling and Fulfillment Foundation table and the `YFS_PERSON_INFO` table. You can create foreign key elements to establish relationship between an extended column and the `YFS_PERSON_INFO` table.

Note: Currently, the YFS_PERSON_INFO is the only table which supports a relationship with foreign key extensions within the Selling and Fulfillment Foundation database.

To add foreign key elements to a standard table:

1. Copy the <INSTALL_DIR>/repository/entity/extensions/Extensions.xml.sample file as <INSTALL_DIR>/extensions/global/entities/<your_filename>.xml file OR modify your existing extension XML file.
2. Edit the <your_filename>.xml file to add foreign key elements as shown in [Example 3–4](#) for each table you want to extend. For a description of the XML attributes, see [Table 3–6](#) on page 3-27.

Example 3–4 Sample XML for Adding Foreign Key Elements

```

<!-- element exposed to create foreign key relationship -->
<DBSchema>
  <Entities>
    <Entity TableName="REQUIRED">
      .
      .
      <!-- element exposed to create relationship with PERSON_INFO table -->
      <ForeignKeys>
        <ForeignKey ParentTableName="YFS_PERSON_INFO"
          XMLName="YFSName1" >
          <Attribute ColumnName="REQUIRED"
            ParentColumnName="PERSON_INFO_KEY" />
        </ForeignKey>
        <ForeignKey ParentTableName="YFS_PERSON_INFO"
          XMLName="YFSName2" >
          <Attribute ColumnName="REQUIRED"
            ParentColumnName="PERSON_INFO_KEY" />
        </ForeignKey>
        .
        .
      </ForeignKeys>
      .
      .
    </Entity>
  </Entities>
</DBSchema>

```

Table 3–6 *Creating ForeignKey elements on YFS_PERSON_INFO table*

Attribute	Description
Entity	
TableName	Required. Name of the table for which the foreign key elements are added; For example: YFS_ITEM.
Entity/ForeignKeys/ForeignKey	
ParentTableName	The name of the parent table for this foreign key element. Note: This value must be YFS_PERSON_INFO which is the only table that currently supports foreign key relationships.
XMLName	You can specify the XML representation of the element name. It must start with the prefix of the parent entity. For example, if ParentTableName is prefixed with YFS then the XMLName must start with YFS. By default the parent table name is assumed.
Entity/ForeignKeys/ForeignKey/Attribute	
ColumnName	Specifies the extended column name of the Entity.
ParentColumnName	The column name of the YFS_PERSON_INFO that has a foreign key element relationship.

3. Create a new `ForeignKey` tag for each foreign key relationship you want to add.
4. Multiple foreign key elements can be related to the same parent table.
5. Extend the corresponding API templates to include the foreign key elements by following the instructions in [Chapter 6, "Extending API Templates"](#).
6. Build and Deploy your extensions.

3.6 Adding Text Search Indices to a Standard Table

You can add text search indices to entities. You add text search indices to a standard Selling and Fulfillment Foundation table, by adding an `TSIndex` element in the extension XML for that table.

To add text search indices to a standard table:

1. Copy the `<INSTALL_DIR>/repository/entity/extensions/Extensions.xml.sample` file as `<INSTALL_DIR>/extensions/global/entities/<your_filename>.xml` file OR modify your existing extension XML file.
2. Edit the `<your_filename>.xml` file to add text search indices as shown in [Example 3–5](#) for each table you want to extend. For a description of the XML attributes, see [Table 3–7](#) on page 3-25.

Example 3–5 Sample XML for Adding Text Search Indices

```
<!-- element exposed to create index -->
<DBSchema>
  <Entities>
    <Entity TableName="REQUIRED">
      .
      .
    <TSIndices>
      <TSIndex Name="REQUIRED" >
        <Column Name="USERNAME" />
      </TSIndex>
      .
    </TSIndices>
    .
  </Entity>
</Entities>
</DBSchema>
```

Table 3–7 Creating Text Search Indices on a Selling and Fulfillment Foundation Table

Attribute	Description
Entity	
TableName	Required. Name of the table for which the text search indices are added. For example: YFS_USER.
Entity/TSIndex	
Name	Required. The name of the text search index. For example: YFS_TS_USER_Name. Note: This value cannot exceed 18 characters.
Entity/TSIndex/Column	
Name	Required. The name of the column for which the text search index is added. You cannot define multiple columns on a text index.

3. Create a new TSIndex tag for each text search index you want to add to the column.
4. Build and Deploy your extensions.

3.7 Enabling Case Insensitive Searches

You can make search operations in the application case insensitive by enabling case insensitive search for the required entity/column. This is achieved by adding the attribute `CaseInsensitiveSearch` in the required entity XML file.

To enable case insensitivity:

1. Edit the required Entity XML to include the `CaseInsensitiveSearch=Y` attribute.

Include the shadow column attribute `ShadowColumnName` in the Entity XML and specify a name for the shadow column. If this is left blank, the system auto generates one. However, you must specify a shadow column name if you are indexing the shadow column.

See [Section 3.7.1, "Modifying an Entity XML"](#) for more information.

A shadow column is then generated, which is linked to the original column that is marked as case insensitive.

2. Specify the configuration mode for the entity or column in the `CaseInsensitiveSearch.Mode` property in the `customer_overrides.properties` file. Set the value of the `CaseInsensitiveSearch.Mode` property to `MIXED` or `DISABLED`. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

Note: By default, case insensitivity is enabled.

3. Disable the cache for the entity or column for which you want to run the Case Insensitive Data Loader agent. For additional information about disabling cache, see the *Selling and Fulfillment Foundation: System Management and Administration Guide*.

Note: Sterling Commerce recommends that along with disabling the cache, you turn off the audits for that particular entity or column as well. For additional information about turning off audits, see the *Selling and Fulfillment Foundation: Extending the Database Guide*

4. Run the Case Insensitive Data Loader agent to populate data in shadow columns. For additional information about Case Insensitive Data Loader agent, see the "Time-Triggered Transaction Reference" appendix in *Sterling Distributed Order Management: Configuration Guide*.
5. Set the value of the `CaseInsensitiveSearch.Mode` property to `ENABLED` in the `customer_overrides.properties` file. For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.
6. Enable the cache for the entity or column for which the Case Insensitive Data Loader agent was run in [Step 3](#). For additional information about disabling cache, see the *Selling and Fulfillment Foundation: System Management and Administration Guide*.

The following APIs support case insensitive searches for system defined columns:

- getOrderList
- getItemListForOrdering
- getExceptionListForOrder
- getCustomerList

For custom columns that have been extended, the case insensitive search works if the normal search works on those columns.

Note: This feature is applicable only for searches that use the application generated XAPIs. Searches using custom queries will not be case insensitive.

Note: Case insensitive search can be enabled only for text data type (Char/Varchar columns).

3.7.1 Modifying an Entity XML

1. Edit the required Entity XML to include the attribute `CaseInsensitiveSearch` under the `<Attributes>` tag, for required columns as shown below.

Example 3–6

```
<Entity Description="This table stores all the exceptions raised by the system."
      EntityType="TRANSACTION" Extensible="Y" Module="ycp"
      Name="Inbox" Prefix="YFS_" TableName="YFS_INBOX" XMLName="Inbox">
  <Attributes>
    .....
    <Attribute ColumnName="EXCEPTION_TYPE"
      DataType="Text-40" DefaultValue="' ' "
      Description="The type of exception."
      Name="Exception_Type" Nullable="false" XMLName="ExceptionType"
```

```
        CaseInsensitiveSearch ="Y"  ShadowColumnName="ExceptionType_LC"/>
        .....
<Indices>
  <Index Name="EXCEPTION_TYPE_I1">
    <Column Name="ExceptionType_LC"/>
  </Index>
</Indices>
```

2. Include the `ShadowColumnName` attribute and specify a name for the shadow column, as shown in Example 2-5.

3.8 Creating Custom and Hang-off Tables

With the Selling and Fulfillment Foundation Database Framework you can also extend the Selling and Fulfillment Foundation database by creating custom or hang-off tables.

A **custom table** is an independent table and cannot be modeled as an extension to a standard Selling and Fulfillment Foundation table.

A **hang-off** table is a table with a many-to-one relationship with a standard Selling and Fulfillment Foundation table.

Creating a custom or hang-off entity enables you to:

- Create a relationship between a Selling and Fulfillment Foundation standard table and a hang-off table.
- Invoke Extensible APIs that store and retrieve data from hang-off tables.
- Invoke `dbverify` for generating appropriate SQL scripts to create or alter tables for custom or hang-off entities.
- Audit item and organization tables.

The ability to purge data from hang-off tables is discussed in [Section 3.8.3](#). Keep in mind the following which apply to the creation of custom or hang-off tables:

- You can only determine if an entity is enabled for hang-off by referencing the associated Entity Relationship Diagram (ERD) located in the `<INSTALL_DIR>/xapidocs/ERD` directory.
- Based on the `Extensions.xml` file, Selling and Fulfillment Foundation does not create a foreign key constraint in the `EFrame_`

TableChanges.sql, but the foreign key relationship is enforced in Selling and Fulfillment Foundation.

- Currently only order, order line, work order, shipment, item, and organization tables are marked as hang-off enabled.
- Custom and hang-off table names must not start with a Y.
- The "Extn" part is trimmed off from the XML name of the custom and hang-off tables.
- Primary key name must not start with a Y.
- Primary key can be of numeric data type.
- Entity names must start with the prefix provided in the entity definition.
- The YIFApi interface does not extend APIs for custom/hang-off tables. Therefore, the APIs for these tables must be configured as services.
- Javadocs are not created for the APIs created by the infrastructure to support custom and hang-off tables.
- XSD generation and validation is not done for custom or hang-off tables.
- Every custom or hang-off entity must have a primary key.

Table 3–8 Required Column for Custom or Hang-off Tables

Column Name	Data Type	Default Value
Key-Column	Key OR Any numeric data type	' ' (space)

- (Optional) A custom or hang-off entity can have the following columns described in [Table 3–9](#):

Table 3–9 Optional Columns for Custom or Hang-off Tables

Column Name	Data Type	Default Value
CREATETS	TimeStamp	sysdate
MODIFYTS	TimeStamp	sysdate
CREATEUSERID	UserId	' ' (space)
MODIFYUSERID	UserId	' ' (space)

Table 3–9 Optional Columns for Custom or Hang-off Tables

Column Name	Data Type	Default Value
CREATEPROGID	ProgramID	' ' (space)
MODIFYPROGID	ProgramID	' ' (space)
LOCKID	Lockid	0 (zero)

3.8.1 Steps to create a custom table

1. Copy the <INSTALL_
DIR>/repository/entity/extensions/Extensions.xml.sample file as <INSTALL_DIR>/extensions/global/entities/<your_filename>.xml file OR modify your existing extension XML file. For example, assume that ABC_CUSTOMER_ORDER_LINE is a custom table.
2. Edit the <your_filename>.xml file to create custom tables as shown in [Example 3–7](#). For a description of the XML attributes, see [Table 3–10](#) on page 3-36.

Example 3–7 Sample XML for Creating Custom Tables

```

<DBSchema>
  <Entities>
    <Entity ApiNeeded="Y/N" AuditRequired="Y" Description=" "
      HasHistory="True/False" Prefix="ABC"
      TableName="ABC_CUSTOMER_ORDER_LINE" >
      <!-- table columns -->
      <Attributes>
        <Attribute ColumnName="CREATETS" DataType="TimeStamp"
          DefaultValue="sysdate" Description="Create TimeStamp" />
        <Attribute ColumnName="MODIFYTS" DataType="TimeStamp"
          DefaultValue="sysdate" Description="Modify TimeStamp" />
        <Attribute ColumnName="CREATEUSERID" DataType="UserId"
          DefaultValue="&apos; &apos;" Description="Creating User ID" />
        <Attribute ColumnName="MODIFYUSERID" DataType="UserId"
          DefaultValue="&apos; &apos;" Description="Modifying User ID" />
        <Attribute ColumnName="CREATEPROGID" DataType="ProgramID"
          DefaultValue="&apos; &apos;" Description="Creating Program ID" />
        <Attribute ColumnName="MODIFYPROGID" DataType="ProgramID"
          DefaultValue="&apos; &apos;" Description="Modifying Program ID" />
        <Attribute ColumnName="LOCKID" DataType="Lockid" DefaultValue="0"
          Description="Lock ID" />
        <Attribute ColumnName="TABLE_KEY" DataType="Key" DefaultValue=" "
    
```

```

        Description="" Nullable="True/False" XMLName="TableKey" />
        .
        .
</Attributes>
<!-- PrimaryKey is a mandatory attribute in entity definition. This
element can have ONLY ONE attribute element -->
<PrimaryKey Name="TABLE_NAME_PK">
    <Attribute ColumnName="TABLE_KEY" />
</PrimaryKey>
<!-- Indices -->
<Indices>
    <Index Name="INDEX_I1" Unique="True/False">
        <Column Name="Attribute2" />
        .
        .
    </Index>
    .
    .
</Indices>
<!-- Relationship -->
<Parent ParentTableName="YFS_ORDER_LINE" XMLName="YFSOrderLine" >
    <Attribute ColumnName="CUSTOM_ORDER_KEY" ParentColumnName="ORDER_
    LINE_KEY" />
    .
    .
</Parent>
<!-- ForeignKeys -->
<ForeignKeys>
    <ForeignKey ParentTableName="PARENT_ORDER_LINE"
    XMLName="PARENTName1" >
        <Attribute ColumnName="CUSTOM_ORDER_KEY" ParentColumnName="PARENT_
        COLUMN_KEY" />
        .
        .
    </ForeignKey>
    .
    .
</ForeignKeys>
<!-- AuditReferences -->
<AuditReferences>
    <Reference ColumnName="TABLE_KEY" />
    .
    .
</AuditReferences>
</Entity>

```

```
</Entities>
</DBSchema>
```

3. The following table explain the attributes in the entity XML:

Table 3–10 Entity XML Definitions for Custom Tables

Attribute	Description
Entity	
ApiNeeded	<p>Indicate whether or not APIs should be generated. Valid values are Y or N. A default set of API's are generated if Y is passed.</p> <p>For example in the ABC_CUSTOMER_ORDER_LINE tables, Selling and Fulfillment Foundation creates the following APIs when the database extension jar file is generated:</p> <pre>getABCCustomerOrderLine() getABCCustomerOrderLineList() createABCCustomerOrderLine() changeABCCustomerOrderLine() deleteABCCustomerOrderLine()</pre> <p>These APIs can be accessed as services using the Service Definition Framework. For more information see, Section 6.4, "Configuring Services for Custom and Hang-off APIs".</p>
AuditRequired	<p>If set to Y audit record for this entity are created. Generating audit for entities is described in Chapter 5, "Generating Audit References for Entities".</p>
HasHistory	<p>This flag indicates whether the custom table can have an history table associated with it.</p> <p>The default value is "False".</p> <p>If the flag is set to "True", the appropriate scripts for generating database scripts for creating and altering the history table is generated by dbverify.</p> <p>For a custom table, the HasHistory flag must be set to "True" for generating history tables. However, if a Parent relationship is defined in the entity XML, this flag is copied from the parent table definition, and all child entities cannot override this flag.</p>

Table 3–10 Entity XML Definitions for Custom Tables

Attribute	Description
Prefix	The prefix added to your custom tables. It is recommended that you do not use a prefix starting with "Y".
TableName	The name given to your custom table.
Entity/Attributes/Attribute	
ColumnName	The names of the column that comprise the table.
DataType	The data type of the column. Valid data types are given in <INSTALL_DIR>/repository/datatypes/datatypes.xml file.
CompressionSupported	Optional. Attribute used to indicate whether or not the data compression is supported for this column. Valid values are True or False. If True compression support is enabled. Note: If the data inserted into a column has both CompressionSupported and UseCompression attributes set to True, then this attribute should not be set back to False. Doing so would result in all compressed data being retrieved without any decompression.
UseCompression	Optional. Attribute used to compress data for this column. Valid values are True or False. If True data is compressed. Note: The value of this attribute should be set to True only if the CompressionSupported attribute is set to True.
QueryAllowed	Optional. Attribute used to enable a compressible column to be used for queries in a List API. Valid values are True or False. If True the column can be used for queries in the List APIs. Note: If the value of the CompressionSupported attribute is set to true then the value of this attribute should also be set to True.
DefaultValue	Default value for the column.
Description	A description of the columns that could be used in javadocs or ERD.

Table 3–10 Entity XML Definitions for Custom Tables

Attribute	Description
Nullable	Optional. Attribute used to describe the nullable value of a field. Default is false. <code>Nullable=true</code> is allowed for all columns except Primary Key Attributes and Entity Relationships.
XMLName	Optional. XML name of the attribute, if it is different from the name of the attribute. Choose a name that does not conflict with the base extension. It is recommended that you use <code>Extn</code> as a prefix. It is also strongly recommended that you use the same convention for arriving at the XMLName as the Selling and Fulfillment Foundation base product does: Make each letter following the underscore in the column name upper case, and the rest lower case. Then, remove the underscores. Thus, <code>Extn_Item_Id</code> should be: <code>ExtnItemId</code> .
Entity/PrimaryKey	
Name	Name of the unique index created for the primary key. This value cannot exceed 18 characters. Note: The name of the primary key in the extension XML should end with <code>_PK</code> .
ColumnName	The name of the table column that is identified as the primary key.
Entity/Indices/Index	
Name	The index name. This value cannot exceed 18 characters.
Unique	This key is present only for custom entities. Valid values are <code>True</code> or <code>False</code> . If <code>True</code> a unique index is created.
AllowCompressedColumns	Optional. If <code>True</code> , the index is allowed to contain columns marked with <code>CompressionSupported</code> attribute set to <code>True</code> .
Column/ Name	The table column name associated with the index.
Entity/Parent	
ParentTableName	Name of the other table this entity has foreign key relationship.

Table 3–10 Entity XML Definitions for Custom Tables

Attribute	Description
XMLName	The XML name of the parent attribute. It should start with the prefix mentioned in the parent table. By default the parent table name is assumed.
Parent/Attribute Level	
ParentColumnName	Column name in the parent table. Note: To create relationships among entities, the data type of parent column must be of type CHAR or VARCHAR.
ColumnName	Column name in this custom entity.
Entity/ForeignKeys/ForeignKey	
ParentTableName	The name of the table with which the entity has a foreign key relationship.
XMLName	XML representation of the element name. By default the parent table name is assumed.
Entity/ForeignKeys/ForeignKey/Attribute	
ParentColumnName	Column name of the parent table. Note: To create foreign keys among entities, the data type of parent column must be of type CHAR or VARCHAR.
ColumnName	Column name in this custom entity.
Entity/AuditReferences/Reference	
ColumnName	Reference Column name in the audit table.

Note: In entity definition, relationship can be defined under `Parent` and `ForeignKey` elements.

4. The relationship defined under the `ForeignKey` element indicates:
 - a. If the foreign table is a Selling and Fulfillment Foundation table, for a single record in the foreign table, zero or many records in this custom table may exist.

- b. This is a read-only relationship, hence deletion of a record from the foreign table does not result in the deletion of a matching record from this custom table.
5. The relationship defined under the `Parent` element indicates:
 - a. For a single record in the parent table, multiple child records may exist.
 - b. Deletion of a record from the parent table results in the deletion of matching records from the child table, if any.
6. Extend the corresponding API templates (for example, `getOrderDetails()` API) by following the instructions in [Chapter 6, "Extending API Templates"](#).

Note: The APIs generated by Selling and Fulfillment Foundation for the custom tables can be invoked only as a service. For more information see [Section 6.4, "Configuring Services for Custom and Hang-off APIs"](#).

7. Build and Deploy your extensions.

3.8.2 Steps to create a hang-off table

1. Copy the `<INSTALL_DIR>/repository/entity/extensions/Extensions.xml.sample` file as `<INSTALL_DIR>/extensions/global/entities/<your_filename>.xml` file OR modify your existing extension XML file. For example, assume that `ABC_CUSTOMER_ORDER_LINE` is a hang-off table.
2. Edit the `<your_filename>.xml` file to create hang-off tables as shown in [Example 3–8](#). For a description of the XML attributes, see [Table 3–11](#) on page 3-36.

Example 3–8 Sample XML for Creating Hang-off Tables

```
<DBSchema>
  <Entities>
    <Entity ApiNeeded="Y/N" AuditRequired="Y" Description=" "
      HasHistory="True/False" Prefix="ABC"
      TableName="ABC_CUSTOMER_ORDER_LINE" >
```

```

<!-- table columns -->
<Attributes>
  <Attribute ColumnName="CREATETS" DataType="TimeStamp"
    DefaultValue="sysdate" Description="Create TimeStamp" />
  <Attribute ColumnName="MODIFYTS" DataType="TimeStamp"
    DefaultValue="sysdate" Description="Modify TimeStamp" />
  <Attribute ColumnName="CREATEUSERID" DataType="UserId"
    DefaultValue="&apos; &apos;" Description="Creating User ID" />
  <Attribute ColumnName="MODIFYUSERID" DataType="UserId"
    DefaultValue="&apos; &apos;" Description="Modifying User ID" />
  <Attribute ColumnName="CREATEPROGID" DataType="ProgramID"
    DefaultValue="&apos; &apos;" Description="Creating Program ID" />
  <Attribute ColumnName="MODIFYPROGID" DataType="ProgramID"
    DefaultValue="&apos; &apos;" Description="Modifying Program ID" />
  <Attribute ColumnName="LOCKID" DataType="Lockid" DefaultValue="0"
    Description="Lock ID" />
  <Attribute ColumnName="TABLE_KEY" DataType="Key" DefaultValue=" "
    Description=" " Nullable="True/False" XMLName="TableKey" />
  .
  .
</Attributes>
<!-- PrimaryKey is a mandatory attribute in entity definition. This
element can have ONLY ONE attribute element -->
<PrimaryKey Name="TABLE_NAME_PK">
  <Attribute ColumnName="TABLE_KEY" />
</PrimaryKey>
<!-- Indices -->
<Indices>
  <Index Name="INDEX_I1" Unique="True/False">
    <Column Name="Attribute2" />
    .
    .
  </Index>
  .
  .
</Indices>
<!-- Relationship -->
<Parent ParentTableName="YFS_ORDER_LINE" XMLName="YFSOrderLine" >
  <Attribute ColumnName="CUSTOM_ORDER_KEY"
    ParentColumnName="ORDER_LINE_KEY" />
  .
  .
</Parent>
<ForeignKeys>
  <ForeignKey ParentTableName="PARENT_ORDER_LINE"

```

```

XMLName="PARENTName1" >
  <Attribute ColumnName="CUSTOM_ORDER_KEY"
    ParentColumnName="PARENT_COLUMN_KEY" />
  .
  .
</ForeignKey>
  .
  .
</ForeignKeys>
<!-- AuditReferences -->
<AuditReferences>
  <Reference ColumnName="TABLE_KEY" />
  .
  .
</AuditReferences>
</Entity>
</Entities>
</DBSchema>

```

3. The following table explain the attributes in the entity XML:

Table 3–11 Entity XML Definitions for Hang-off Tables

Attribute	Description
Entity	
ApiNeeded	<p>Indicate whether or not APIs should be generated. Valid values are Y or N. A default set of API's are generated if Y is passed.</p> <p>For example in the ABC_CUSTOMER_ORDER_LINE table, Selling and Fulfillment Foundation creates the following APIs when the database extension jar file is generated:</p> <pre>listABCCustomerOrderLine() getABCCustomerOrderLine() createABCCustomerOrderLine() modifyABCCustomerOrderLine() deleteABCCustomerOrderLine()</pre> <p>These APIs can be accessed as services using the Service Definition Framework. For more information see, Section 6.4, "Configuring Services for Custom and Hang-off APIs".</p>

Table 3–11 Entity XML Definitions for Hang-off Tables

Attribute	Description
AuditRequired	<p>If set to <code>Y</code> audit record for this entity are created. Generating audit for entities is described in Chapter 5, "Generating Audit References for Entities".</p> <p>Note: This attribute must not be passed when you are creating a hang-off for order related tables. In this case, the audits are automatically inserted into the <code>YFS_ORDER_AUDIT</code> table.</p>
Description	A description of the entity that could be used in javadocs or ERD.
HasHistory	<p>This flag is automatically inherited from the parent table. For example, let us assume that <code>ABC_ORDER_HEADER</code> table is created as an hang-off table for <code>YFS_ORDER_HEADER</code>, which has an associated history table. Then <code>ABC_ORDER_HEADER_H</code> is automatically generated by the database framework.</p>
Prefix	The prefix added to your custom tables. It is recommended that you do not use a prefix starting with <code>Y</code> .
TableName	The name given to your hang-off table.
Entity/Attributes/Attribute	
ColumnName	The names of the column that comprise the table.
DataType	The data type of the column. Valid data types are given in <code><INSTALL_DIR>/repository/datatypes/datatypes.xml</code> file.
CompressionSupported	<p>Optional. Attribute used to indicate whether or not the data compression is supported for this column. Valid values are <code>True</code> or <code>False</code>. If <code>True</code> compression support is enabled.</p> <p>Note: If the data inserted into a column has both <code>CompressionSupported</code> and <code>UseCompression</code> attributes set to <code>True</code>, then this attribute should not be set back to <code>False</code>. Doing so would result in all compressed data being retrieved without any decompression.</p>

Table 3–11 Entity XML Definitions for Hang-off Tables

Attribute	Description
UseCompression	<p>Optional. Attribute used to compress data for this column. Valid values are True or False. If True data is compressed.</p> <p>Note: The value of this attribute should be set to True only if the CompressionSupported attribute is set to True.</p>
QueryAllowed	<p>Optional. Attribute used to enable a compressible column to be used for queries in a List API. Valid values are True or False. If True the column can be used for queries in the List APIs.</p> <p>Note: If the value of the CompressionSupported attribute is set to true then the value of this attribute should also be set to True.</p>
DefaultValue	Default value for the column
Description	A description of the columns that could be used in javadocs or ERD.
Nullable	<p>Optional. Attribute used to describe the nullable value of a field. Default is false. Nullable=true is allowed for all columns except Primary Key Attributes and Entity Relationships.</p>
XMLName	<p>Optional. XML name of the attribute, if it is different from the name of the attribute.</p> <p>Choose a name that does not conflict with the base extension. It is recommended that you use Extn as a prefix. It is also strongly recommended that you use the same convention for arriving at the XMLName as the Selling and Fulfillment Foundation base product does: Make each letter following the underscore in the column name upper case, and the rest lower case. Then, remove the underscores. Thus, Extn_Item_Id should be: ExtnItemId.</p>
Entity/PrimaryKey	
Name	<p>Name of the unique index created for the primary key. This value cannot exceed 18 characters.</p> <p>Note: The name of the primary key in the extension XML should end with <code>_PK</code>.</p>

Table 3–11 Entity XML Definitions for Hang-off Tables

Attribute	Description
ColumnName	The name of the table column that is identified as the primary key.
Entity/Indices/Index	
Name	The index name. This value cannot exceed 18 characters.
Unique	This key is present only for custom entities. Valid values are True or False. If True a unique index is created.
AllowCompressedColumns	Optional. If True, the index is allowed to contain columns marked with CompressionSupported attribute set to True.
Column/ Name	The table column name associated with the index.
Entity/Parent	
ParentTableName	Name of the other table this entity has foreign key relationship.
XMLName	The XML name of the parent attribute. It should start with the prefix mentioned in the parent table. By default the parent table name is assumed.
Parent/Attribute Level	
ParentColumnName	Column name in the parent table. Note: To create relationships among entities, the data type of parent column must be of type CHAR or VARCHAR.
ColumnName	Column name in this custom entity.
Entity/ForeignKeys/ForeignKey	
ParentTableName	The name of the table with which the entity has a foreign key relationship.
XMLName	XML representation of the element name. By default the parent table name is assumed.
Entity/ForeignKeys/ForeignKey/Attribute	

Table 3–11 Entity XML Definitions for Hang-off Tables

Attribute	Description
ParentColumnName	Column name of the parent table. Note: To create foreign keys among entities, the data type of parent column must be of type CHAR or VARCHAR.
ColumnName	Column name in this hang-off entity.
Entity/AuditReferences/Reference	
ColumnName	Reference Column name in the audit table.

Note: In entity definition, relationship can be defined under `ForeignKey` elements.

4. The relationship defined under the `ForeignKey` element indicates:
 - a. If the foreign table is a Selling and Fulfillment Foundation table, for a single record in the foreign table, zero or many records in this hang-off table may exist.
 - b. This is a read-only relationship, hence deletion of a record from the foreign table does not result in the deletion of a matching record from this hang-off table.
5. Extend the corresponding API templates (for example, `getOrderDetails` API) by following the instructions in [Chapter 6, "Extending API Templates"](#).

Note: The APIs generated by Selling and Fulfillment Foundation for the hang-off tables can be invoked only as a service. For more information see [Section 6.4, "Configuring Services for Custom and Hang-off APIs"](#)

6. Build and Deploy your extensions.

3.8.3 Purging Data from Hang-Off Tables

Currently, the Purge agent moves records to history tables. With the custom or hang-off entities enabled, the Purge agent also deletes records from hang-off tables. However, the data from a hang-off table can be purged only if its parent elements are also purged. If a history table exists, records are added to the history table. The records are deleted from the history table using the History Purge agent.

In order to purge the custom and hang-off entities you need to include the `entities.jar` file in the classpath of the agent server. For more information on setting up an agent server, see the *Selling and Fulfillment Foundation: Installation Guide*.

Extending the Data Types Files

4.1 Extending the Data Types Files

You can extend the attributes available to you by adding your own XML attributes and abstract data types to the `datatypes.xml` file.

4.1.1 Extending the Data Type Map File

To extend the data type map XML file:

1. Create a new `<INSTALL_DIR>/extensions/global/template/resource/yfsdatatypepemap.xml` file.

If the `/global/template/resource/` directory structure does not exist, create the required directory structure.

2. Add an XML root node in the same way it appears in the `<INSTALL_DIR>/repository/xapi/template/merged/resource/yfsdatatypepemap.xml` file.
3. Add any attributes that need to be mapped in the `yfsdatatypepemap.xml` file.

4.1.2 Extending the Data Type File

To extend the data type XML file:

1. Create a new `<INSTALL_DIR>/extensions/global/etc/datatypes.xml` file.

If the `/global/etc/` directory structure does not exist, create the required directory structure.

2. Add an XML root node in the same way it appears in the `<INSTALL_DIR>/repository/datatypes/datatypes.xml` file.
3. Add any differential values for the datatypes, including the following:
 - Add and define parameters for new datatypes
 - Modify parameters of existing datatypes

Note: For existing datatypes, you can modify only the UI related attributes in the `datatypes.xml` file such as UI Size, and UI TableSize.

Note: You cannot resize the Date input fields within the Application Console across the board even if your date format is larger than the default date format used by Selling and Fulfillment Foundation.

Note: Selling and Fulfillment Foundation reserves the `Type` attribute for internal use, and so you cannot override it. All other attributes can be overridden.

Generating Audit References for Entities

5.1 About Extending Audit References

If the `AuditRequired` flag is enabled in the entity XML, audit records are added to the `YFS_AUDIT` table. The default for this flag is `Y`, for item and organization tables. However, the audit flag and audit references can be overridden by the extension XML file.

Note: All the records pertaining to the cached tables as well as the tables for which the value of `AuditRequired` flag is set to `Y` are logged into `YFS_AUDIT` table.

If you want to switch off the generation of audit references for some specific entities, change the value of the `AuditRequired` flag to `N` for such entities.

Note: You can add new audit references in the extension XML file. When new references are added, they take precedence over the existing audit references, which are entirely overridden.

You can add up to six audit references only.

Only item and organization header-level audit records are inserted in the `YFS_AUDIT_HEADER` table. The audit references refer to the columns of the entity being audited.

The audits can be generated for the hang-off and custom tables, by modifying the entity table name and audit reference column names.

Note: Auditing is not supported for hang-off tables with more than one parent.

To generate audit references for entities:

1. Edit the <your_filename>.xml file in the <INSTALL_DIR>/repository/entity/extensions directory to enable audit record generation for desired entities. The following example explains the elements to be added to the database schema:

Example 5–1 Sample XML for Creating Audit References

```
<DBSchema>
  <Entities>
    <Entity TableName="YFS_ITEM" AuditRequired="Y" >
      .
      .
      <AuditReferences>
        <Reference ColumnName="ItemId" />
        .
        .
      </AuditReferences>
      .
      .
    </Entity>
  </Entities>
</DBSchema>
```

Table 5–1 Generating audits for entities

Attribute	Description
Entity	
TableName	The table name to be audited.
AuditRequired	If this flag is set to Y the audit references are entered in the YFS_AUDIT table. Note: This attribute must not be passed when you are creating a hang-off for order related tables. In this case, the audits are automatically inserted into the related order audit tables.

Table 5–1 *Generating audits for entities*

Entity/AuditReferences/Reference	
ColumnName	The column name in this entity which has audit references. This name must be valid for the entity.

2. Create a new `Reference` tag for each audit reference you want to add.
3. The hang-off of an order table audits can be viewed with the associated order audits.

Extending API Templates

6.1 About Extending API Templates

Each template-based API delivers different output, depending on the template passed to it. To verify whether an API is template-based or not, see the *Selling and Fulfillment Foundation: Javadocs*.

If your table modifications impact any APIs, you must extend the templates of those APIs. Place the extended API templates in the `<INSTALL_DIR>/extensions/global/template/api` directory.

To find out which APIs are impacted by table modifications:

1. Note the `XMLName` attribute of the table being modified in the `entity` tag inside the database entity XML files (which contains the definition of all the tables). These database entity XML files are located in `<INSTALL_DIR>/repository/entity` directory.
2. Search for the pattern of that `XMLName` attribute in the `<INSTALL_DIR>/extensions/global/template/api` directory. The search results in finding exposed and internal APIs impacted by the table modifications or extensions.

For example, consider that you want to extend an attribute in the `YFS_CHARGE_CATEGORY` table. The `XMLName` for this table as specified in `<INSTALL_DIR>/repository/entity/omp_tables.xml` is `ChargeCategory`. Now search for the attribute `ChargeCategory` in `<INSTALL_DIR>/extensions/global/template/api` directory to find the APIs impacted by this extension.

6.2 Including Extended Attributes in the API Template

The extended attributes appear as a separate `<Extn>` element under the primary element.

For example, in the default output XML template of the `getItemDetails()` API, the `Item` attributes have the following structure:

```
<?xml version="1.0" encoding="UTF-8"?>
  <Item .. Item attributes >
    <PrimaryInformation .... PrimaryInformation attributes />
    <ItemServiceSkillList .. ItemServiceSkillList attributes/>
    <ItemAliasList ... ItemAliasList attributes />
    .
    .
  </Item>
```

After extending the `Item` header, the `getItemDetails()` API can output the following XML:

```
<?xml version="1.0" encoding="UTF-8"?>
  <Item .. Item attributes >
    <PrimaryInformation .... PrimaryInformation attributes />
    <Extn ExtnAltQty="200408201034469490" ..... extnded attributes />
    <YFSPersonInfo .... PersonInfoKey="200408201034469490" ..... />
    <ItemServiceSkillList .. ItemServiceSkillList attributes/>
    <ItemAliasList ... ItemAliasList attributes />
    .
    .
  </Item>
```

Note: Foreign Key variables for the extended column appear as a `PersonInfoKey` attribute of the `YFSPersonInfo` element. The relationship can be validated if the extended column and the `PersonInfoKey` have the same value.

The extended attribute is retrieved from the `XMLName` attribute of the `<your_filename>.xml` file that you edited in the previous sections, when extending a standard table. Place your extended templates in the `<INSTALL_DIR>/extensions/global/template/api` directory.

6.3 Including Custom and Hang-Off Entities in the API Template

The standard APIs can be extended to provide information from the custom or hang-off tables. A tool specifically provided for generating the template XML's, `templateXmlGen.xml` is located in the `<INSTALL_DIR>/bin` directory.

1. Run the template XML generation tool from your `<INSTALL_DIR>` directory by using the following command:

```
sci_ant.sh -Dtable=<TABLE_NAME> -f bin/templateXmlGen.xml
```

2. Once the command is run, the sample XML files are placed in the `<INSTALL_DIR>/extensions/global/webpages/sampleXML` directory as `<TABLE_NAME>_sample.xml`.

For example, consider `HF_Order_Header` is an hang-off of `YFS_Order_Header` table. The generated `HF_Order_Header_sample.xml` is as follows:

```
<HFOrderHeader Createprogid=" " Description=" " DocumentType=" "
EnterpriseKey=" "OrderHeaderKey=" " OrderName=" " OrderNo=" " .... >
</HFOrderHeader>
```

3. A sample XML for including the above attributes in a standard API can be generated by passing the `YFS` table that has relationship with the hang-off table you are interested.

For example, assume `HF_Order_Header` is a hang-off table with a relationship to the `YFS_Order_Header` table. The XML template generated by the tool when `TABLE_NAME=YFS_Order_Header` is passed:

```
<Order>
  <OrderLines>
    <OrderLine .....>
      <Extn extended attributes >
        <HFOrderHeaderList>
          <HFOrderHeader Createprogid=" " Description=" " ..... >
          </HFOrderHeader>
        </HFOrderHeaderList>
      </Extn>
    </OrderLine>
```

```
</OrderLines>  
</Order>
```

Note: You can modify the attributes only within your custom or hang-off element.

You can prune this sample XML to include your custom attributes in an API template, such as `getOrderDetails` output template. However, you cannot modify any of the YFS elements or attributes.

Note: The sample XMLs are also automatically generated when you create the database extension jar file, and are posted in the `<INSTALL_DIR>/xapidocs/sampleXML` directory. However, if you need to create a sample template as described in [Step 3](#), you must run the template XML generation tool separately by specifying the corresponding YFS table name.

4. An hang-off table can be deleted by passing an `Operation` attribute in the `change` or `modify` APIs. For example, `HF_Order_Header` element can be deleted in a `changeOrder` API as:

```
<Order>  
  <OrderLines>  
    <OrderLine .....>  
      <Extn extended attributes >  
        <HFOrderHeaderList>  
          <HFOrderHeader Operation="Delete" Createprogid=" " ..... >  
        </HFOrderHeader>  
      </HFOrderHeaderList>  
    </Extn>  
  </OrderLine>  
</OrderLines>  
</Order>
```

The operations such as **Create** and **Modify** are run by default. If an entry for that element exists, the API modifies the entries with the recent value. In the case where that element does not exist it creates a new entry.

- The records in a hang-off table can be reset by assigning the value of "true" to the Reset attribute at the list level element of the XML. When the records are reset, all existing records for that hang off table that correspond to the parent table are deleted and all elements included under the list element are inserted. For example, hang off records in the HF_Order_Header_list element can be reset by using the following:

```

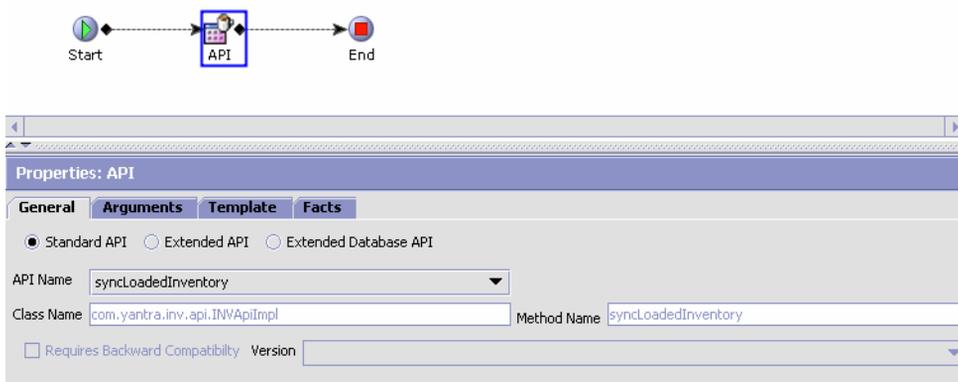
<Order>
  <OrderLines>
    <OrderLine .....>
      <Extn extended attributes >
        <HFOrderHeaderList Reset="true">
          <HFOrderHeader>
            </HFOrderHeader>
          </HFOrderHeaderList>
        </Extn>
      </OrderLine>
    </OrderLines>
  </Order>

```

- Build and Deploy your extensions.

6.4 Configuring Services for Custom and Hang-off APIs

The APIs generated for custom or hang-off entities by Selling and Fulfillment Foundation can be invoked only as a service. The service configuration user interface has to be enabled to configure these APIs. For more information on creating a service definition see the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.



For including custom APIs you can create a service definition as shown in the figure. The configuration fields are explained in [Table 6–1](#).

Table 6–1 API Configuration Properties

Field Name	Description
General Tab	
Standard Selling and Fulfillment Foundation API	Select this option if a standard Selling and Fulfillment Foundation API is to be invoked. If selected, a Standard Selling and Fulfillment Foundation API Name drop down list displays. For each API, the Class Name and Method Name are provided and cannot be edited.
Extended API	Select this option if a custom java code is to be invoked.

Table 6–1 API Configuration Properties

Field Name	Description
Extended Database API	<p>Select this option if the service invokes a custom or hang-off API. If selected, a custom API Name drop-down list displays. For each API, the Class Name and Method Name are provided and cannot be edited.</p> <p>Note: If you want to lock a record in a custom table, pass the <code>SelectMethod</code> attribute as part of the input XML to the custom entity API. The locking happens within the transaction boundary of the custom API call.</p> <p>The <code>SelectMethod</code> attribute can take the following values:</p> <p>WAIT, NO_WAIT, and NONE.</p> <p>For more information about locking records in Extended APIs, refer to the <i>Selling and Fulfillment Foundation: Customizing APIs Guide</i></p>
API Name	<p>Select or enter the API to be called.</p> <p>Note: This field is for integration purposes only.</p>
Class Name	Specifies the class to be called.
Method Name	Specifies the method to be called.
Requires Backward Compatibility	Select this field to indicate that input data coming through the API is from a previous version (only applicable to Selling and Fulfillment Foundation system APIs).
Version	If you chose Requires Backward Compatibility, select the Selling and Fulfillment Foundation version the API is to behave as. Only the applicable versions for the individual API display.

Table 6–1 API Configuration Properties

Field Name	Description
Arguments Tab	
Argument Name	You can pass name/value pairs to the API by entering the values in the Arguments Tab. In order for custom APIs to access custom values, the API should implement the interface <code>com.yantra.interop.japi.YIFCustomApi</code> . If entered, these name/value pairs are passed to the CustomApi as a Properties object.
Argument Value	Enter the argument value.
Template Tab	
	When the Selling and Fulfillment Foundation System APIs are invoked, you can specify an output template to be used by the API. You can specify the template in the configuration properties of the Service Definition, the Resource Definition in the Resource Hierarchy tree, or both. However, if the template has been specified at both definition levels, the template specified in the Service Definition is used.
XML Template	Select this radio button to construct the XML to be used for the API output. Enter the template root element name and click OK. You can then construct the XML.
File Name	Select this radio button to enter the filename of the XML file to be used as the API output template. This file should also exist in your CLASSPATH.
Facts Tab	
	A Fact is an attribute that is used by an API or an agent in Selling and Fulfillment Foundation to identify which colony to connect to and retrieve data from. Based on the fact name and fact value entered, the corresponding colony is determined.

Table 6–1 API Configuration Properties

Field Name	Description
Fact Name	Enter the fact name of the XML attribute.
Fact Value	Enter the fact value of the XML attribute.

A

APIs

- extending templates, 53
- including extended attributes, 54
- including hang-off tables, 55

audit

- entity audit generation, 51

C

columns

- adding, 4
- adding data compression support, 5
- extending, 4, 5

custom tables

- creating, 35

customization checklist, 3

customizing

- data type map XML files, 49
- data type XML file, 49

D

data type map XML files, 49

data type XML file, 49

database extensibility

- adding custom table, 33
- adding foreign key elements, 27
- adding hang-off tables, 33
- adding unique descriptors, 23
- adding unique tag identifiers, 23

Descriptors, 23

E

Entity Relationship Diagrams, 3

Extending API templates, 53

extensible files

- yfsdatatypepemap.xml, 49

F

foreign keys

- adding, 7

G

GuideAddNonUniqueIndicesToStdTable, 6

guidelines

- adding foreign key elements, 7
- adding support for data compression in columns, 5
- extending columns, 4
- modifying tables, 3

H

hang-off tables

- creating, 42
- purging data, 48

M

mobile user interfaces

- creating custom, 3, 49, 51

N

nullable columns, 4

T

Tag identifiers, 23