Sterling Commerce
An AT&T Company

Selling and Fulfillment Foundation: Localization Guide

Release 9.0

March 2010

Copyright Notice

STERLING COMMERCE SOFTWARE

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

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. Erik Arvidsson. Copyright © 2008 Azer Koçulu http://azer.kodfabrik.com. Copyright © Einar Lielmanis, einars@gmail.com. Copyright © 2006 John Reilly (www.inconspicuous.org) and Copyright © 2002 Douglas Crockford (www.crockford.com). Copyright © 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 © Microsoft Corp. 1981-1998. Purple Technology, Inc. Copyright © 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 © 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 © 2007, the OWASP Foundation. Copyright Raghu K, 2003. Copyright © 2004 David Schweinsberg. Copyright © 2005-2006 Darren L. Spurgeon. Copyright © 2005-2008 Sam Stephenson. Copyright © S.E. Morris (FISH) 2003-04. Copyright © 1998 Regents of the University of California. Copyright © 2006 VisualSoft Technologies. Copyright © 2002-2009 Zipwise Software. All rights reserved by all listed parties.

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

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 IMIPLIED WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTIBILITY, AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTIAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERUPTION) 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, IlligalArgumentException.java and MissingOptArgException.java software was originally based on software copyright © 2001, Sun Microsystems, http://www.sun.com. For more information on the Apache Software Foundation, please s

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: installdir/jar/commons_upload/1_0/ CommonsFileUpload_License.txt,
installdir/jar/jetspeed/1_4/RegExp_License.txt,
installdir/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/9.0/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 © 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 © 1999, IBM Corporation., http://www.ibm.com.
- software copyright © 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 © 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 © 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 © 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.

ANTISAMY SOFTWARE

The Sterling Commerce Software is distributed with or on the same storage media as the AntiSamy software (Copyright © 1998 Regents of the University of California. All rights reserved.) (the "AntiSamy Software"). The AntiSamy Software is free software which is distributed under the terms of the following license:

Copyright © 1998, Regents of the University of California

All rights reserved.

Redistribution and use 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 the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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 HOLDER 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.

COOLBUTTONS SOFTWARE

The Sterling Commerce Software is also distributed with or on the same storage media as Coolbuttons.js ("Coolbuttons Software"), which is subject to the following license:

This Button Script was designed by Erik Arvidsson for WebFX. For more info and examples see: http://webfx.eae.net or send email to erik@eae.net. Feel free to use this code as long as this disclaimer is intact.

The preceding license only applies to the Coolbuttons Software and does not apply to the Sterling Commerce Software, or 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.

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

THE ECLIPSE SOFTWARE FOUNDATION

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

com.ibm.icu.nl1_3.4.4.v200606220026.jar, org.eclipse.ant.core.nl1_3.1.100.v200606220026.jar,
org.eclipse.ant.ui.nl1_3.2.0.v200606220026.jar, org.eclipse.compare.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.boot.nl1_3.1.100.v200606220026.jar,
org.eclipse.core.commands.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.contenttype.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.expressions.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.filebuffers.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.filesystem.nl1_1.0.0.v200606220026.jar,
org.eclipse.core.jobs.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.resources.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.runtime.compatibility.auth.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.runtime.compatibility.nl1_3.1.100.v200606220026.jar,
org.eclipse.core.runtime.nl1_3.2.0.v200606220026.jar,
org.eclipse.core.variables.nl1_3.1.100.v200606220026.jar,
org.eclipse.debug.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.debug.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.equinox.common.nl1_3.2.0.v200606220026.jar,
org.eclipse.equinox.preferences.nl1_3.2.0.v200606220026.jar,
org.eclipse.equinox.registry.nl1_3.2.0.v200606220026.jar,
org.eclipse.help.appserver.nl1_3.1.100.v200606220026.jar,
org.eclipse.help.base.nl1_3.2.0.v200606220026.jar, org.eclipse.help.nl1_3.2.0.v200606220026.jar,
org.eclipse.help.ui.nl1_3.2.0.v200606220026.jar, org.eclipse.jdt.apt.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.apt.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.core.manipulation.nl1_1.0.0.v200606220026.jar,
org.eclipse.jdt.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.debug.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.doc.isv.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.doc.user.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.junit4.runtime.nl1_1.0.0.v200606220026.jar,
org.eclipse.jdt.launching.nl1_3.2.0.v200606220026.jar, org.eclipse.jdt.nl1_3.2.0.v200606220026.jar,
org.eclipse.jdt.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.jface.databinding.nl1_1.0.0.v200606220026.jar,
org.eclipse.jface.nl1_3.2.0.v200606220026.jar, org.eclipse.jface.text.nl1_3.2.0.v200606220026.jar,
org.eclipse.ltk.core.refactoring.nl1_3.2.0.v200606220026.jar,
org.eclipse.ltk.ui.refactoring.nl1_3.2.0.v200606220026.jar,
org.eclipse.osgi.nl1_3.2.0.v200606220026.jar, org.eclipse.osgi.services.nl1_3.1.100.v200606220026.jar,
org.eclipse.osgi.util.nl1_3.1.100.v200606220026.jar, org.eclipse.pde.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.pde.doc.user.nl1_3.2.0.v200606220026.jar,
org.eclipse.pde.junit.runtime.nl1_3.2.0.v200606220026.jar,
org.eclipse.pde.nl1_3.2.0.v200606220026.jar, org.eclipse.pde.runtime.nl1_3.2.0.v200606220026.jar,
org.eclipse.pde.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.platform.doc.isv.nl1_3.2.0.v200606220026.jar,
org.eclipse.platform.doc.user.nl1_3.2.0.v200606220026.jar,
org.eclipse.rcp.nl1_3.2.0.v200606220026.jar, org.eclipse.search.nl1_3.2.0.v200606220026.jar,
org.eclipse.swt.nl1_3.2.0.v200606220026.jar, org.eclipse.team.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.team.cvs.core.nl1_3.2.0.v200606220026.jar,
org.eclipse.team.cvs.ssh.nl1_3.2.0.v200606220026.jar,
org.eclipse.team.cvs.ssh2.nl1_3.2.0.v200606220026.jar,
org.eclipse.team.cvs.ui.nl1_3.2.0.v200606220026.jar, org.eclipse.team.ui.nl1_3.2.0.v200606220026.jar,
org.eclipse.text.nl1_3.2.0.v200606220026.jar, org.eclipse.ui.browser.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.cheatsheets.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.console.nl1_3.1.100.v200606220026.jar,
org.eclipse.ui.editors.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.externaltools.nl1_3.1.100.v200606220026.jar,
org.eclipse.ui.forms.nl1_3.2.0.v200606220026.jar, org.eclipse.ui.ide.nl1_3.2.0.v200606220026.jar,
org.eclipse.ui.intro.nl1_3.2.0.v200606220026.jar, org.eclipse.ui.navigator.nl1_3.2.0.v200606220026.jar,

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

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

A copy of the Eclipse Foundation Software User Agreement is found at
<install_dir>/SI/repository/rcp/rcpdependencies/windows/eclipse/notice.html,
<install_dir>/SI/repository/rcp/rcpdependencies/windows/eclipse/plugins/notice.html,
<install_dir>/SI/repository/rcp/rcpdependencies/gtk.linux_x86/eclipse/notice.html, and
<install_dir>/SI/repository/rcp/rcpdependencies/gtk.linux_x86/eclipse/plugins/notice.html.

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

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

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

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

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

Ehcache Software

The Sterling Commerce Software is also distributed with or on the same storage media as the ehache v.1.5 software (Copyright © 2003-2008 Luck Consulting Pty. Ltd.) ("Ehache 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/smcfs/9.0/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."

ESAPI SOFTWARE

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

as the ESAPI software (Copyright © 2007, the OWASP Foundation) ("ESAPI Software"). ESAPI Software Software is free software which is distributed under the terms of the following license:

Copyright © 2007, The OWASP 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:

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 the OWASP Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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.

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 © 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.

JAVASCRIPT MINIFIER

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

JSMin.java 2006-02-13; Updated 2007-08-20 with updates from jsmin.c (2007-05-22)

Copyright © 2006 John Reilly (www.inconspicuous.org)

This work is a translation from C to Java of jsmin.c published by Douglas Crockford. Permission is hereby granted to use the Java version under the same conditions as the jsmin.c on which it is based.

jsmin.c 2003-04-21

ICE SOFTWARE

JBOSS SOFTWARE

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 note. 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 disclaimers 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.

JSDoc Tookit Software

The Sterling Commerce Software is distributed with or on the same storage media as the JSDoc Toolkit software (Copyright © 2008 Michael Mathews) ("JSDoc Toolkit Software"), which is subject to the following license:

All code specific to JsDoc Toolkit are free, open source and licensed for use under the X11/MIT License.

JsDoc Toolkit is Copyright © 2008 Michael Mathews <micmath@gmail.com>

This program is free software; you can redistribute it and/or modify it under the terms below.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice must be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

JSLib Software

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

product (Copyright © 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/eardata/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/

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."

Prototype Software

The Sterling Commerce Software is distributed with or on the same storage media as the Prototype software (Copyright © 2005-2008 Sam Stephenson) ("Prototype Software"), which is subject to the following license:

Purple Technology

Rico Software

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 © 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>/jar/rhino/1_7R1/License.txt. 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

SLF4J Software

The Sterling Commerce Software is also distributed with or on the same storage media as the SLF4J software (Copyright © 2004-2008) ("SLF4J Software"), which is subject to the following license:

Copyright © 2004-2008 QOS.ch All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,

INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE 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 -installdir/jar/comm/2_0
SUN ACTIVATION JAR -installdir/jar/jaf/1_0_2
SUN JavaMail -installdir/jar/javamail/1_4

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 <install>/repository/eardata/platform/war/WEB-INF/web_app_License.txt.

The source code for the Web-App Software may be found at:http://java.sun.com/dtd/.

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

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

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

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

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

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

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

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

WARRANTY DISCLAIMER

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

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

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

Contents

Preface

This manual provides an outline for localizing Selling and Fulfillment Foundation.

Intended Audience

This manual is intended to provide a general localization procedure for users responsible for localizing Selling and Fulfillment Foundationfor new languages, as well as languages that are provided by Selling and Fulfillment Foundation. Prior to reading this document, be sure that you have read the *Selling and Fulfillment Foundation: Installation Guide* for knowledge and understanding of the Selling and Fulfillment Foundation database installation.

Structure

This manual contains the following sections:

Chapter 1, "Internationalization in Selling and Fulfillment FoundationSelling and Fulfillment Foundation"

This chapter briefly lists and describes how Selling and Fulfillment Foundation has been internationalized for localization, and lists the components that cannot be localized.

Chapter 2, "Localizing Selling and Fulfillment Foundation"

This chapter describes the Selling and Fulfillment Foundation components that can be localized, and provides general guidelines on how to localize each component.

Chapter 3, "Localizing the Rich Client Platform Application"

This chapter describes and provides general guidelines on how to localize the Selling and Fulfillment Foundation-based Rich Client Platform applications.

Appendix A, "Localizable XML Attributes"

This chapter lists the localizable factory setup XML attributes.

Selling and Fulfillment Foundation Documentation

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

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

- *Selling and Fulfillment Foundation: Customizing Swing Interface Guide*
- *Selling and Fulfillment Foundation: Extending the Condition Builder Guide*
- *Selling and Fulfillment Foundation: Extending the Database Guide*
- *Selling and Fulfillment Foundation: Extending Transactions Guide*
- *Selling and Fulfillment Foundation: Using Sterling RCP Extensibility Tool Guide*
- *Selling and Fulfillment Foundation: Integration Guide*
- *Selling and Fulfillment Foundation: Product Concepts Guide*
- *Sterling Warehouse Management™ System: Concepts Guide*
- *Selling and Fulfillment Foundation: Application Platform Configuration Guide*
- *Sterling Distributed Order Management™: Configuration Guide*
- *Sterling Supply Collaboration: Configuration Guide*
- *Sterling Global Inventory Visibility™: Configuration Guide*
- *Catalog Management™: Configuration Guide*
- *Sterling Logistics Management: Configuration Guide*
- *Sterling Reverse Logistics™: Configuration Guide*
- *Sterling Warehouse Management System: Configuration Guide*
- *Selling and Fulfillment Foundation: Application Platform User Guide*
- *Sterling Distributed Order Management: User Guide*
- *Sterling Supply Collaboration: User Guide*
- *Sterling Global Inventory Visibility: User Guide*
- *Sterling Logistics Management: User Guide*
- *Sterling Reverse Logistics: User Guide*
- *Sterling Warehouse Management System: User Guide*
- *Selling and Fulfillment Foundation: Mobile Application User Guide*
- *Selling and Fulfillment Foundation: Business Intelligence Guide*

- *Selling and Fulfillment Foundation: Javadocs*
- *Sterling Selling and Fulfillment Suite™: Glossary*
- *Parcel Carrier: Adapter Guide*
- *Visual Modeler™: Application Guide*
- *Selling and Fulfillment Foundation: Multitenant Enterprise Guide*
- *Selling and Fulfillment Foundation: Password Policy Management Guide*
- *Selling and Fulfillment Foundation: Properties Guide*
- *Catalog Management: Concepts Guide*
- *Selling and Fulfillment Foundation: Pricing Concepts Guide*
- *Selling and Fulfillment Foundation: Setting Up Quotes*
- *Sterling Sensitive Data Capture Server, Release 1.0: Configuration Guide*
- *Sterling Sensitive Data Capture Server, Release 1.0: PA-DSS Implementation Guide*
- *Selling and Fulfillment Foundation: Secure Deployment Guide*
- *Business Center: Item Administration Guide*
- *Business Center: Pricing Administration Guide*
- *Business Center: Customization Guide*
- *Business Center: Localization Guide*

Conventions

The following conventions may be used in this manual:

Convention	Meaning
. . .	Ellipsis represents information that has been omitted.
< >	Angle brackets indicate user-supplied input.

Convention	Meaning
`mono-spaced text`	Mono-spaced text indicates a file name, directory path, attribute name, or an inline code example or command.
/ or \	Slashes and backslashes are file separators for Windows, UNIX, and Linux operating systems. The file separator for the Windows operating system is "\" and the file separator for UNIX and Linux systems is "/". The UNIX convention is used unless otherwise mentioned.
<INSTALL_DIR>	User-supplied location of the Selling and Fulfillment Foundation installation directory. This is only applicable for Release 8.0 and later.
<INSTALL_DIR_OLD>	User-supplied location of the Selling and Fulfillment Foundation installation directory (for Release 8.0 and later). **Note:** This is applicable only for users upgrading from Release 8.0 and later.
<SSDCS_DIR>	User-supplied location of the Sterling Sensitive Data Capture Server installation directory. This is applicable for Selling and Fulfillment Foundation, Release 9.0 and later.
<YANTRA_HOME>	User-supplied location of the Sterling Supply Chain Applications installation directory. This is only applicable for Releases 7.7, 7.9, and 7.11.
<YANTRA_HOME_OLD>	User-supplied location of the Sterling Supply Chain Applications installation directory (for Releases 7.7, 7.9, or 7.11). **Note:** This is applicable only for users upgrading from Releases 7.7, 7.9, or 7.11.
<YFS_HOME>	For Releases 7.3, 7.5, and 7.5 SP1, this is the user-supplied location of the Sterling Supply Chain Applications installation directory. For Releases 7.7, 7.9, and 7.11, this is the user-supplied location of the <YANTRA_HOME>/`Runtime` directory. For Release 8.0 and later, the <YANTRA_HOME>/`Runtime` directory is no longer used and has been substituted with the location <INSTALL_DIR>.

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

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

- Yantra is used for Release 7.7 and earlier.
- Sterling Supply Chain Applications is used for Releases 7.9 and 7.11.
- Sterling Multi-Channel Fulfillment Solution is used for Releases 8.0 and 8.2.
- Selling and Fulfillment Foundation is used for Releases 8.5 and 9.0.

1

Internationalization in Selling and Fulfillment FoundationSelling and Fulfillment Foundation

This chapter provides an overview of the concepts pertaining to localization and internationalization. It also explains how localization and internationalization apply to Selling and Fulfillment Foundation.

1.1 Localizable Components and Data

This section describes the internationalized components and data that can be localized in your system.

Every user created within Selling and Fulfillment Foundation can have an associated preference for number formatting, date layout, and language. These preferences are called locale, and are identified as a pairing of a language code and a country code. Examples include en_US (English, US), fr_CA (French, Canada), fr_FR (French, France). A specific locale definition includes the following information:

- Country
- Language
- Date and time format
- Time zones
- Numeric Formats
- Currency
- Unit of measure for dimensions, volume, and weight

To enable the association of data such as dates, times, and strings with locale-specific formats, Selling and Fulfillment Foundation associates a locale with each user profile. This allows each user to use a locale-specific version of the product.

The locale definition associated with any organization defined in Selling and Fulfillment Foundation is used to determine only the currency and unit of measure. Date, time, and time zone information is strictly related to each user.

1.1.1 Literals and Data

All user interface and exception message literals are retrieved from a set of external files or database tables.

Selling and Fulfillment Foundation retrieves images and literals from locale-specific files. In order to provide a single-installation multilingual solution, Selling and Fulfillment Foundation stores multiple instances of the literals for a screen. Each instance is identified by a specific country and language pairing.

1.1.2 Multibyte Character Sets

Multi-byte character sets are appropriately and thoroughly taken into consideration in the database, application server, and browser tiers of Selling and Fulfillment Foundation. To represent all the characters in a language, it is sometimes necessary to use 2 (double byte) or 3 (multi-byte) bytes for each character. The longer character representations can, however, pose space and transmission challenges during application development.

- Double Byte Character Set (DBCS): One of a number of character sets defined for representing Chinese, Japanese, or Korean text (for example, JIS X 0208-1990). These character sets are often encoded in such a way as to allow double-byte character encoding to be mixed with single-byte character encoding.
- Multibyte Character Set (MBCS): A character set encoded with a variable number of bytes for each character. Many large character sets have been defined as multi-byte character sets in order to keep strict compatibility with the standards of the ASCII subset, the ISO and IEC 2022.

The Selling and Fulfillment Foundation architecture ensures that:

- All data is stored in the database using a standard compression algorithm known as UTF-8.
- The application is coded in Java, which can handle multi-byte character sets without any special changes.
- All communication between the database and the application server is through Java Database Connectivity (JDBC), which transforms the UTF-8 database representation of data to and from the multi-byte character set.
- All communication between the application server and the client is through UTF-8, which minimizes data transmission volume.
- All clients are expected to receive and send data using the UTF-8 algorithm.

1.1.3 Date and Time Formats

Selling and Fulfillment Foundation can present stored dates and times in any valid date or time format. Date and time fields in Selling and Fulfillment Foundation must be entered relative to the locale in which the Selling and Fulfillment Foundation database resides. Some typical date formats are as follows:

- MM/dd/yyyy
- dd/MM/yyyy
- yyyy/MM/dd

> **Note:** In each date format, the month can be entered as a word instead of a numeral as long as the entire length is not more than ten characters.

Some typical time formats are as follows:

- HH:mm:ss
- HH:mm

Note: It is recommended that you use the hour format 'HH' as opposed to 'hh', whenever you localize Selling and Fulfillment Foundation. 'HH' signifies the 24-hour format, which is the only format supported by Selling and Fulfillment Foundation.

1.1.4 Time Zones

Besides being sensitive to local time zone considerations, Selling and Fulfillment Foundation is configured to recognize worldwide time zones. For example, if an order is placed in Germany for fulfillment in the United States, but, the order details are not filled on time, the software considers differences in the two time zones in order to raise an exception at the appropriate hour in the United States.

Following are the different ways in which the Application handles time zones:

Note: When you define a locale with a time zone (for example, en_US_EST) and when the user is logged in for that locale, the system uses the files ending in <language>_<country> code only. For instance, if the application had time zones fr_CA_EST and fr_CA_GMT set up for a locale, the system uses the same files (for example, ycpapibundle_fr_CA.properties, en_US_ ycplocalizedstrings_fr_CA.properties, earth_fr_CA.css) irrespective of the time zone of the logged in user.

- When date and time values are stored, it is converted to the locale of the database. For example, assume that a database is in New York, and a customer service representative is in London. The customer service representative enters the details of an order. When the order's date and time are stored in the database that resides in New York, the values are converted to Eastern Standard Time.
- When date and time fields are displayed in the user interface, Selling and Fulfillment Foundation performs time zone calculations based on the current locale of the user, and displays the time accordingly. For example, when a customer service representative in London views an

order that resides in a database in New York, the date and time are converted from Eastern Standard Time to Greenwich Mean Time.

> **Note:** When a date field does not contain a time component, the time is assumed to be 12 a.m. Such fields are not adjusted for time zones when viewed from various locales with different time zones.

- The Selling and Fulfillment Foundation APIs display the date and time the way they are stored in the database. For time-sensitive fields, the time zone difference from the Universal Time Coordinate (UTC) is appended to the date and time in the output.

1.1.5 Numeric Formats

Numeric formats are dependent on the country and language set up in the locale definition.

1.1.6 Currency

Selling and Fulfillment Foundation allows each order to be processed in the preferred currency of the customer. This currency is referred to as "transactional currency".

A currency is also associated with the locale that is associated with each enterprise. This is referred to as "enterprise currency".

Order management capabilities provide order handling in multiple currencies between buyer and seller. These capabilities include a multi-currency view of an order's value as well as currency conversion procedures and rate tables.

Data structures hold flexible charge and tax taxonomies for order and invoice entities. Selling and Fulfillment Foundation has deliberately *not* taken the approach of building complex taxation rules into the application. Integration with sophisticated tax calculation programs, such as Taxware or Vertex, complement our solution and provide you with a complete taxation system. Selling and Fulfillment Foundation optionally provides a standard integration to the Taxware product line.

Note: All charges and taxes display in the user interface in the order (transactional) currency. Users can switch to view payment information of an order in either the transactional or enterprise currency.

1.1.6.1 Currency Precision

Selling and Fulfillment Foundation contains the following currency precision features:

- Unit price and unit cost can be entered and stored with a maximum of six decimal places.
- Totals are entered and stored with a maximum of two decimal places.
- Totals are rounded off (through the traditional rounding concept) if they contain 5 digits or more.

Note : When installing Selling and Fulfillment Foundation pack for Release 8.0, you need to create your own currency or you may encounter some errors. For more information on creating currencies, see the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

1.1.6.2 Currency Conversion

Selling and Fulfillment Foundation contains the following currency conversion features:

- Currency conversion rates are defined between two currencies and are bound by effective dates as shown in the example provided in the following table:

Table 1–1 Currency Conversion Rates

From	To	From	To	Rate
British Pound	US Dollar	1/1/2004	1/31/2004	1.51
British Pound	US Dollar	2/1/2004	2/28/2004	1.56
British Pound	US Dollar	3/1/2004	3/31/2004	1.62

A conversion rate definition is understood to imply a 1:x relationship. Thus, if one British pound is equal to 1.51 US dollars during January 2004, two British pounds are equal to 3.02 US dollars during January 2004.

- Rate definitions are not reciprocal. A conversion from US dollars to British pounds cannot use the inverse of the British pound to US dollar exchange rate. A rate for converting US dollars to British pounds must be available. Otherwise, Selling and Fulfillment Foundation reports an error. The only exception to this restriction is that a reciprocal relation does exist between the euro and its member currencies.
- Selling and Fulfillment Foundation provides the updateConversionRates API that allows users to import exchange rates and maintain the exchange rates in their database. Using this API, new exchange rates can be loaded every few hours if desired. You can also set up conversions using the Applications Manager.
- Selling and Fulfillment Foundation provides a user exit at the point of performing a currency conversion. This user exit allows users to perform currency conversions outside of Selling and Fulfillment Foundation. This allows real-time application of the current exchange rate instead of using the last updated exchange rate.

1.1.6.3 Currency Conversion Scenario

The following scenario illustrates how currency is displayed for an order. Suppose that a customer enters an order in France with an English company. The customer pays in francs (the transactional currency). Selling and Fulfillment Foundation converts the amount from francs to pounds using triangulation through the euro.

Because the order is placed in francs, if a company employee in England displays the order, the order is displayed in francs.

To allow auditing or reviewing of currency conversions, Selling and Fulfillment Foundation stores the exchange rates for each order as an order attribute. When an order is entered, the software stores the exchange rate used between the transactional currency and the enterprise currency. If a conversion involves triangulation, Selling and Fulfillment Foundation stores the composite exchange rate used between the euro, the transactional currency, and the enterprise currency.

1.1.7 Units of Measure

A Selling and Fulfillment Foundation user can select different units of measure (UOM) from various places in the Selling and Fulfillment Foundation user interface.

For supported carriers, pack dimensions entered by a shipper can be converted to different dimensions needed by the carrier. For example, if a shipper enters pack dimensions in kilos, kilos can be converted to pounds for a carrier. To accomplish this conversion for other carriers, custom coding is needed.

1.2 Nonlocalizable Components and Data

Selling and Fulfillment Foundation does not support the localization of components that are explicitly technical in nature. These components are typically used by professionals who are working with a Sterling Commerce employee to perform the installation tasks, tuning tasks, and so on. The components include documentation and the following tools:

- Configuration Deployment Tool
- VT220 Mobile Terminal

Note: You can localize your Local Documentation Library implementation, and the Selling and Fulfillment Foundation documentation set.

However, Sterling Commerce does not provide localization support for the Online Documentation Library and the Local Documentation Library.

For more information about the Documentation Library, see the *Selling and Fulfillment Foundation: Installation Guide*.

1.3 Localizing Selling and Fulfillment Foundation Using Language Packs

You can localize Selling and Fulfillment Foundation in a single base language with one or multiple language packs. The base language refers to the display language of the factory setup data in the Applications Manager. For more information about switching the base language, see Section 2.4.1.2, "Switching the Selling and Fulfillment Foundation Base Language".

Note: For language pack installations on Windows XP, where the base language is switched to a non-English language on an English operating system, the following operating system settings must be applied:

1. Navigate to Start > Settings > Control Panel > Regional and Language Options.
2. Under Regional Options, select the applicable language.
3. Depending on the language that you are localizing, under the Languages tab, select the Install files for East Asian languages check box.
4. Click the Advanced tab and select the applicable language.

Note: For language pack installations on Linux, you must install the language fonts. These can be obtained from the installation disks. For example, to install Chinese fonts, install the `zh_CN-2.14-6.noarch.rpm` file.

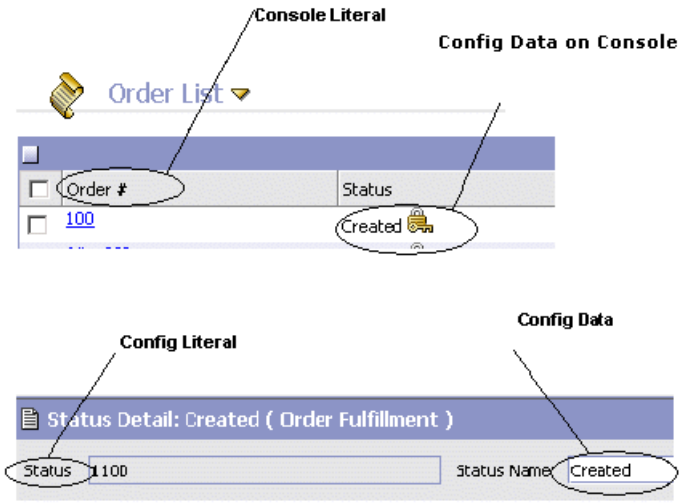
Note: The languages listed in Table 1–2, "Localization Options for Single-Language CD" are for informational purposes only. For a complete list of available languages, contact your Sterling Commerce Accounts Executive.

The following list provides a description of the terms used in the tables:

- Base Language - Refers to the language used to display factory setup data in the Applications Manager.
- Config Literals - Refers to the labels on the Applications Manager screen.
- Console Literals - Refers to the labels on the Application Console screen.
- Config Data - Refers to the factory setup data. A specific list of fields can be translated. For example, the Status Description can be localized.
- Config Data on Console - Refers to the display of the configuration data on the console user interface. For example, configuration data pertaining to the carrier is displayed on the console screen.

Figure 1–1, "Schematic Diagram" represents the terms described in the previous list:

Figure 1–1 Schematic Diagram



The following sections provide more information about localized literals and data for both a single language CD and multiple language CDs.

Single-Language CD

Table 1–2, "Localization Options for Single-Language CD" provides two different options for the localization information if you get a single language pack along with a standard base language.

Table 1–2 Localization Options for Single-Language CD

Options	User Locale	Config Literals	Config Data	Console Literals	Config Data on Console
Option 1: Base Language English with Japanese Language Pack	**Japanese**	Japanese	English	Japanese	Japanese
	English	English	English	English	English
Option 2: Base Language Japanese with English Language Pack	**Japanese**	Japanese	Japanese	Japanese	Japanese
	English	English	Japanese	English	English

Multiple-Language CDs

Each language pack is shipped as a separate CD. If you have, for example, purchased a Japanese Language Pack and Chinese Language Pack, you will have two language CDs.

Table 1–3, "Localization Options for Multiple-Language CD" provides three localization options if you buy multiple language packs along with a standard base language.

Note: Each of the option specified in the table is supported as part of your implementation design. However, you must decide on the best possible option based on your business requirements.

Table 1–3 Localization Options for Multiple-Language CD

Options	User Locale	Config Literals	Config Data	Console Literals	Config Data on Console
Option 1: Base Language Japanese with Chinese and English Language Pack	**Japanese**	Japanese	Japanese	Japanese	Japanese
	English	English	Japanese	English	English
	Chinese	Chinese	Japanese	Chinese	Chinese

Table 1–3 Localization Options for Multiple-Language CD

Options	User Locale	Config Literals	Config Data	Console Literals	Config Data on Console
Option 2: Base Language Chinese with Japanese and English Language Pack	**Japanese**	Japanese	Chinese	Japanese	Japanese
	English	English	Chinese	English	English
	Chinese	Chinese	Chinese	Chinese	Chinese
Option 3: Base Language English with Japanese and Chinese Language Pack	**Japanese**	Japanese	English	Japanese	Japanese
	English	English	English	English	English
	Chinese	Chinese	English	Chinese	Chinese

2

Localizing Selling and Fulfillment Foundation

This chapter provides steps for localizing Selling and Fulfillment Foundation for both languages that are not yet provided and languages that are supported by Selling and Fulfillment Foundation out of the box.

2.1 Prerequisites to Localizing your Application

Before you begin localizing Selling and Fulfillment Foundation, read the information provided in this section.

Rules Governing Localization

Some of the localization rules that Selling and Fulfillment Foundation follows are governed by standards and rules external to Selling and Fulfillment Foundation, such as Java Virtual Machine (JVM), Java Runtime Environment (JRE), operating systems, and external applications. The Java programming language specifies the implementation of the locale logic. For example, see http://java.sun.com/j2se/1.5.0/docs/api/java/util/ResourceBundle.html for more information about the logic around resource bundles. For more information about specifying a JRE, see Chapter 2.2.1, "Specifying the JRE Settings".

Note: It is recommended that an exact match be used for all localization files (_<lang>_<country> in the file name) to ensure that all the files are correctly returned.

International Standards Organization Codes

Selling and Fulfillment Foundation uses locale-related codes as specified by the International Standards Organization (ISO). Throughout this guide, locale is represented by <language>_<country>. Locale comprises of <language>, a 2-character lower-case ISO-639 code, and <country>, a 2-character upper-case ISO-3166 code.

Character Encoding

When planning the localization of Selling and Fulfillment Foundation, ensure that your operating system default character encoding is set to that specified by the yfs.ui.defaultEncoding property in the yfs.properties file during product installation.

To modify this property, add an entry for it in the <INSTALL_DIR>/properties/customer_overrides.properties file. For additional information about overriding properties using the customer_overrides.properties file, see the *Selling and Fulfillment Foundation: Properties Guide*.

To enable specification of characters other than the ASCII set (for example, Asian characters and vowels with accents), ensure that the WebSphere JVM client.encoding.override property is set to UTF-8, when deploying on IBM WebSphere. For more information about the required WebSphere JVM settings, see the *Selling and Fulfillment Foundation: Installation Guide*.

Literals External to Selling and Fulfillment Foundation

External applications used in conjunction with Selling and Fulfillment Foundation can display on-screen information. For example, Oracle WebLogic displays a status message during startup. Non-Selling and Fulfillment Foundation literals of this type can be localized. For instructions about how to localize these types of messages, see the documentation supplied by the corresponding software provider.

Microsoft Windows Clients

When localizing the Applications Manager to run on Windows, ensure that client computers have regional options set appropriately in the Windows Control panel. For example, if the client computer is running Windows and you want to display the title “Applications Manager" in Japanese, the client Windows regional options must be set to Japanese.

When localizing the Application Console to run on Windows, ensure that client computers can correctly display Unicode characters, by configuring the display from Control Panel > Display > Appearance.

Mobile Device Clients

The `ycpapibundle.properties` and `yscpapibundle.properties` files contains literals that apply specifically to mobile devices. The key for these mobile device literals are preceded with "`Mobile_`". When localizing Selling and Fulfillment Foundation to run on a mobile device, localize the literals associated with these keys.

2.2 Setting Up Character Encoding

When implementing Selling and Fulfillment Foundation, use the guidelines specified in this section when using Latin-1 and UTF-8 encoding.

2.2.1 Specifying the JRE Settings

In a multilanguage deployment, install the international version of the JRE on each client computer.

To install the plugin, download the JRE from http://java.sun.com/javase/downloads/index_jdk5.jsp and install it on each client computer. For clients that must display non-English literals, select the "Windows (all languages, including English)" JRE.

2.2.2 Latin-1 Character Encoding

ISO 8859-1 is the ISO standard Latin-1 character set and encoding format. CP1252 is what Microsoft defined as the superset of ISO 8859-1. Thus, there are approximately 27 extra characters that are not included in the standard ISO 8859-1.

When using Latin-1 character encoding, it is recommended that you also refer to the following documents:

- For more information about Basic Latin and the Latin-1 Supplement that together comprise the Latin-1 (or ISO 8859-1) character set and encoding, see http://www.unicode.org/charts/.
- For an official chart download, see:

  http://www.iso.ch/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=28245&ICS1=35&ICS2=40&ICS3= ().
- For a Microsoft site that lists both ISO 8859-1 and CP1252, see:

  http://www.microsoft.com/globaldev/reference/WinCP.mspx.

2.2.3 UTF-8 Character Encoding

Encoding must be specified for international characters to be displayed correctly on UI components. For example:

- Inventory Graph
- Delivery Map
- Shipnode in Inventory Console
- Item ID in Inventory Console

On UNIX, the UTF-8 character encoding must be specified on your UNIX application server.

To set up a UTF-8 environment:

1. Modify all process startup scripts (for the application server, time-triggered transactions, and so on) to include the `-Dfile.encoding=UTF-8` parameter for the Java command.
2. Alternatively, when working on a UNIX application server, determine whether or not it has UTF-8 capability by running the following command:

   ```
   locale -a
   ```

- If the command returns *any* line that indicates UTF-8, proceed to Step 1.

  For example:

  ```
  POSIX
  common
  en_US.UTF-8
  C
  iso_8859_1
  ```

- If the command does not return any lines that indicate UTF-8, proceed to Step 3.

3. From the international language option pack appropriate to your UNIX operating system, install at least one language that has the UTF-8 character set, and then return to Step 2 to test the installation.

2.3 Database Overview

You must create the Application database suitable to the data encoding format and character set used. The size of the database fields also depends on the data encoding format and type of character sets used. The following sections explain the installation and setting up of Oracle, SQL and DB2 databases.

Data Encoding Format

Selling and Fulfillment Foundation is tested and shipped using the UTF-8 transformation format. If you use a different transformation encoding format, the number of characters that you can store in standard sized database diminishes. In this case, ensure that you review and modify the Selling and Fulfillment Foundation database creation process in order to size the database fields accordingly.

Character Set

Use a character set appropriate for your localization language. For example, single-byte language character sets typically require UTF-8, while a multi-byte language may require a UTF-16 character set.

The character set you choose may impact field sizes. For example, a Varchar(40) field can only store 40/3 Japanese characters using the UTF-8 character set. This has implications on the table field sizes at the

time of creation. Table creation scripts must be modified to ensure that the field lengths are correct.

Note: For the Japanese locale, the AL32UTF-8 character set or the UTF-16 character set must be used.

String Length Checker

To run the string length checker for ensuring that the translated strings do not exceed the field lengths of the tables, perform the following steps:

1. Create a folder named `/Length`.
2. Copy the contents of `<INSTALL_DIR>/repository/entity`, including all subfolders, into `/Length/entity`.
3. Copy the contents of `<INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS` into `/Length/XMLS`.

   Note: The factorysetup directory is generated when the COPY_FCXML_TO_REPOSITORY property is set to True. For information about COPY_FCXML_TO_REPOSITORY, see the *Selling and Fulfillment Foundation: Properties Guide*.

4. Copy the `<INSTALL_DIR>/repository/datatypes/datatypes.xml` into `/Length`.
5. Copy the following JAR files to the `/Length/lib` directory:
   - `<INSTALL_DIR>/jar/platform_afc/5_0/platform_afc.jar`
   - `<INSTALL_DIR>/jar/install_foundation.jar`
   - `<INSTALL_DIR>/jar/log4j/1_2_15/log4j-1.2.15.jar`
   - `<INSTALL_DIR>/jdk/jre/lib/endorsed/xalan.jar`
   - `<INSTALL_DIR>/jdk/jre/lib/endorsed/xercesImpl.jar`
   - `<INSTALL_DIR>/jdk/jre/lib/endorsed/xml-apis.jar`
6. Copy `<INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS/<language>_<country>/<baselanguage>_`

<basecountry>_<prefix>localizedstrings_<language>_<country>.properties to /Length

7. Set CLASSPATH=/Length/lib/xml-apis.jar;/Length/lib/xalan.jar;/Length/lib/xercesImpl.jar;/Length/lib/platform_afc.jar;/Length/lib/install_foundation.jar;/Length/lib/log4j-1.2.15.jar

8. Run the following Java command. This command runs the string length checker in GENERATE mode. In this mode, the output file contains a list of translatable literals and their maximum string lengths.

   ```
   call <JAVA_HOME>/bin/java com.yantra.ycp.tools.localization.YCPLocalizedStringLengthTool -OUTPUT_FILE /Length/LengthsFile.txt -MODE GENERATE -ENTITY_DIR /Length/entity -DTYPES_FILE /Length/datatypes.xml -FC_DIR /Length/XMLS
   ```

9. Run the following java command. This command runs the string length checker in CHECK mode. In this mode, the localizedstrings file (for instance en_US_ycplocalizedstrings_ja_JP.properties) is compared with the LengthsFile.txt file that is generated from running the string length checker in GENERATE mode. Running this tool in CHECK mode also creates MissingLength.txt, which contains the literals that are missing from LengthsFile.txt, and MissingTranslations.txt, which contains the literals missing from the localizedstrings file that was passed in the input.

   ```
   call <JAVA_HOME>/bin/java com.yantra.ycp.tools.localization.YCPLocalizedStringLengthTool -OUTPUT_FILE inconsistencies.txt -MODE CHECK -LENGTHS_FILE /Length/LengthsFile.txt -TRANSLATIONS_FILE /Length/<baselanguage>_<basecountry>_<prefix>localizedstrings_<language>_<country>.properties.
   ```

Note: Unicode is recognized as one character. For example, On\ Order\ Release\ Status\ Change is translated as Changer de statut lors du d\u00e9bloquage de la commande.The number of characters in the converted text is fifty. If you search for On\ Order\ Release\ Status\ Change, in the `LengthsFile.txt` file, the following is displayed:

On\ Order\ Release\ Status\ Change=50

This indicates that the maximum permissible length is fifty. The length of translated text exceeds the maximum permissible length by one.

Factory Defaults

Factory defaults are limited to one language. This means that in order to switch from one language to another, you must load new factory defaults to your database.

The UI literals are not a part of the factory defaults. They can be switched from one language to another as required, as long as they have been translated appropriately.

When using an Oracle database, it is possible to use Japanese characters in English factory defaults. In fact, it could be any character in valid Unicode and UTF-8 range. When using a Microsoft SQL Server database, it is not possible to use more than one encoding and code page, based on the database collation chosen at time of the database creation. Therefore, if the database is created with a collation that is not Japanese, the factory defaults cannot contain Japanese character collation, such as Latin1 (SQL_Latin1_General_CP1_CI_AS).

2.3.1 Oracle Database Setup

In this release of Selling and Fulfillment Foundation, multilingual (including a multi-byte character set) environments with Oracle databases have been tested and certified using Oracle 10g created with a UTF-8 character set. See the *Selling and Fulfillment Foundation: Installation Guide* for more information about installing Oracle 10g and the settings for the UTF-8 character set.

2.3.2 Microsoft SQL Server Database Setup

Microsoft SQL Server has a limitation pertaining to the collation and code page with which the database is created to store characters. Because of this, during database creation, you must carefully consider collation issues pertaining to storage of data with non-English international characters, including supported Asian code pages.

It is recommended that a Microsoft SQL Server has case-insensitive collation (specified using the CI argument). For example, to select a case-insensitive collation for the Japanese language, one of the valid collation choices is Japanese_CI_AI.

The localizable factory setup is limited to the selected collation and code page because they are stored in database.

Microsoft SQL Server permits one locale for each database.

To set up Microsoft SQL Server:

1. From the Microsoft SQL Server collation name drop-down menu, select the collation name that supports the language you want to specify.Ensure that you select a collation that is case insensitive. This creates your database.
2. Configure the `charset` value for the JDBC URL property to match the code page of the collation name selected in Microsoft SQL Server in the `<INSTALL_DIR>/properties/customer_overrides.properties` as shown below:

   Jdbc url format is
   "jdbc:inetdae7:<hostName>?charset=<yourCharset>"

   For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.
3. Install the application tables and other components, and then load the database factory defaults, as described in the *Selling and Fulfillment Foundation: Installation Guide*.

2.3.3 DB2 Database Setup

DB2 database can be localized by passing the codeset as UTF-8. For more information about the DB2 database setup, see the *Selling and Fulfillment Foundation: Installation Guide*.

2.4 Data Storage

Besides storing your transactional data, the database also stores configuration data, such as error codes and item descriptions of various attributes. This means that the database may have to store values in a language-specific format. If these database literals are not localized, screen literals are displayed inconsistently, with some of them being displayed in the localized language and the others in English.

In addition to localizing configuration data, Selling and Fulfillment Foundation, Release 8.5, enables you to localize master data, such as items, categories, and assets.

You can store item descriptions in your database in multiple languages. For more information about item descriptions, see the *Catalog Management: Configuration Guide*.

Note: The localized item description is only available for Order Details and Order Line Details screens.

2.4.1 Localization for a Multilanguage Installation

The database factory default values can be localized for a multi-language installation, to enable the use of one or more locales in addition to the installed locale.

To localize the factory default XML and user-configured attributes for multiple locales:

1. Create the necessary locale using the Applications Manager. For more information about creating a locale, see the *Platform User Guide for the Console Interface Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
2. Run the LocalizedStringReconciler tool from `<INSTALL_DIR>/bin` in either the extract_errorcause_db mode or the EXPORT mode for UNIX

and Windows as described here. Running it in the extract_errorcause_db mode extracts the error cause and error action literals.

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml export
-Ddestdir=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml
export -Ddestdir=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS
```

This processes the entity XMLs and identifies the missing literals for the localizable columns in the YFS_LOCALIZED_STRINGS table.

The data is then exported to one or more properties files depending on the number of locales present in the YFS_LOCALE table.

To export for a specific locale, use optional parameters language, and country. This usage is as follows:

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml export
-Ddestdir=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS -Dlanguage=fr -Dcountry=FR
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml export
-Ddestdir=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS -Dlanguage=fr -Dcountry=FR
```

Note: In an instance where the YFS_LOCALE table contains two entries other than the base locale, two files are created. For example, if fr_FR (representing French, and France) is the locale, a file named en_US_ycpmissinglocalizedstrings_fr_FR.properties is created in the destination folder specified (destdir).

3. Edit the aa_BB_ycpmissinglocalizedstrings_xx_YY.properties file for the relevant locale, in the <INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS/<xx_YY> directory, where aa is the language code for the "From" locale, BB is the country code for the "From" locale, xx is the language code for the "To" locale, and YY is the country code for the "To" locale.

   The aa_BB_ycpmissinglocalizedstrings_xx_YY.properties file contains entries in the following format:

   ```
   Acceptance_Process=
   Add_Service=
   Line=
   ```

   Add relevant translation values to the entries. Following is an example for French literals:

   ```
   Acceptance_Process=Processus d'acceptation
   Add_Service=Ajouter service
   Line=Ligne
   ```

4. Save the modified aa_BB_ycpmissinglocalizedstrings_xx_YY.properties file in the escaped Unicode format. In this file, all the multi-byte characters should be in escaped Unicode format. The following example displays the requisite formatting for lines in this file:

   For the Japanese locale:
   `Add\Line=\u660e\u7d30\u306e\u8ffd\u52a0`

   For the French locale:
   `Invalid\inventory\operation.=Op\u00e9ration d'inventaire non valide.`

5. Run the LocalizedStringReconciler tool from <INSTALL_DIR>/bin in IMPORTTEST mode as described here. This ensures that there are no problems during the import process.

   For UNIX:

   ```
   sci_ant.sh -f localizedstringreconciler.xml importtest
   -Dsrc=<INSTALL_DIR>/repository/factorysetup/complete_
   installation/XMLS
   ```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml importtest
-Dsrc=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS
```

You can also specifically provide the Dbasefilename parameter in the localizedstringreconciler.xml file. For example:

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml importtest
-Dsrc=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS -Dbasefilename=ycpmissinglocalizedstrings
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml importtest
-Dsrc=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS -Dbasefilename=ycpmissinglocalizedstrings
```

Note: The IMPORTTEST functionality of the LocalizedStringReconciler tool helps you to test the import process and roll back the changes, if required. The IMPORTTEST mode enables you to test the import process and verify if any strings cannot be added to the appropriate fields in the database for any reason, for example, extensive length.

This is particularly important when you are upgrading from a previous release because, after a base language is switched, the import process will place the translation into the database.

The function of IMPORTTEST is similar to SWITCHTEST, in that, just as SWITCHTEST helps a new customer verify whether the switch process will be successful, IMPORTTEST helps a customer who is upgrading to verify whether the import process will be successful.

6. Run the LocalizedStringReconciler tool from <INSTALL_DIR>/bin in IMPORT mode as described here. This inserts the values specified in the properties file into the database.

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml import
-Dsrc=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml import
-Dsrc=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS
```

You can also specifically provide the Dbasefilename parameter in the localizedstringreconciler.xml file. For example:

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml import
-Dsrc=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS -Dbasefilename=ycpmissinglocalizedstrings
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml import
-Dsrc=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS -Dbasefilename=ycpmissinglocalizedstrings
```

Note: By default, when the localizedstringreconciler tool is run without passing the -Dbasefilename parameter, the tool picks up and runs the ycplocalizedstrings file, which contains the Selling and Fulfillment Foundation Factory Setup.

Note: For customers who are upgrading from previous versions, some error codes will not be translated when a new language pack is applied, as error codes will not change and as such will not be in the latest language pack.

To resolve this, you have to

- Run the localizedstringreconciler tool in export mode
- Translate old/custom error codes
- Run the tool in import mode to have your translations added in the database.

2.4.1.1 Extending the Default Factory-Shipped Translations

The default factory-shipped translations can be extended to create custom translations for the localization literals.

To modify the default factory-shipped translations with custom localization literals:

1. Create a new extn folder in the <INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS/<language>_<country> directory.

   Note: For example, if fr_FR is the factory-shipped translated locale, the extn folder should be created in the <INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS/fr_FR directory.

2. Copy the <baselanguage>_<basecountry>_ycplocalizedstrings_<language>_<country>.properties file from the <INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS/<language>_<country> directory to the newly created extn folder.
3. Edit the translations in the properties file in the extn folder:
   - Modify the translations for existing localization literals with the new translations.

- Add new localization literals and their translations, if required.
- Remove any obsolete or unwanted translations that are not overridden for localization literals.

4. Run the LocalizedStringReconciler tool from <INSTALL_DIR>/bin in IMPORT mode as follows:

   For UNIX:

   ```
   sci_ant.sh -f localizedstringreconciler.xml import -Dsrc=<INSTALL_
   DIR>/repository/factorysetup/complete_installation/XMLS
   ```

   For Windows:

   ```
   sci_ant.cmd -f localizedstringreconciler.xml import -Dsrc=<INSTALL_
   DIR>\repository\factorysetup\complete_installation\XMLS
   ```

   This tool first inserts the values specified in the properties file present in the <INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS/<language>_<country> directory into the database.

   This entry is then replaced with the values specified in the properties file in the <INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS/<language>_<country>/extn directory.

2.4.1.2 Switching the Selling and Fulfillment Foundation Base Language

The base language is the language that all translations are in relation to. For example, when a new common code is added, the description is in the base language, and there are translations from this language to other languages.

Note: The base language can be switched only **once**. Switching a base language more than once may result in loss of data or other potential errors.

When this switch is performed, the following processes occur:

- Every translation to the desired base language from your current base language is changed to a translation from your desired base language to your current base language.
- Every translation from your current base language to another language is changed to a translation from your desired base language to another language.
- Every localizable column is converted from your current base language to your desired base language.

For example, your Selling and Fulfillment Foundation install has English as a base language, and you have set up translations to French and German. The translations are interpreted as follows:

- Translations from English to French
- Translations from English to German

By switching your base language to French, your translations are interpreted as follows:

- Translations from French to English
- Translations from French to German

Note: In order to successfully perform a base language switch, all the localizable descriptions from the current base language to the desired base language must exist. Furthermore, there must be at least one entry in the YFS_LOCALE table corresponding to the desired base language.

2.4.1.2.1 Considerations When Switching the Base Language

The following points must be taken into consideration prior to performing a base language switch:

- Multiple "from" strings in a language may translate to the same "to" string in another language. During the switch, only one of these records is retained, and ambiguous records are removed.

  For example, ABC and AAC in English may both translate to FABC in French. When the base language is switched from English to French, only one record with a "from" string of FABC is retained.

- If a locale does not provide a translation, Selling and Fulfillment Foundation reverts to the base language.

  For example, there are three locales, en_US, en_GB, and fr_FR. Currently en_US is the base language. fr_FR has full translations, but en_GB only has a few translations, such as *color* to color. If the en_GB locale is used and the en_GB translation for *apple* does not exist, Selling and Fulfillment Foundation reverts to the en_US translation of *apple*.

  When the base language is switched to French, it becomes the fallback language. So, if the en_GB translation of *apple* does not exist, Selling and Fulfillment Foundation falls back on the fr_FR translation, *pomme.*

  It is recommended that *all* translations be provided for *all* locales.

Note: When you switch the base language, the translated values may exceed the database column size, depending on the database character set (UTF-8, AL16UTF16 or any other) used. To correct the problem, adjust the translated characters to fit the database column. It is recommended that you use a character set that is appropriate to your localization requirements.

To switch the base language, perform the following tasks:

1. To ensure that there are no problems in making a base language switch, run the LocalizedStringReconciler tool from <INSTALL_DIR>/bin in SWITCHTEST mode as follows:

   For UNIX:

   ```
   sci_ant.sh -f localizedstringreconciler.xml switchtest
   -Dlanguage=xx -Dcountry=YY
   ```

   For Windows:

   ```
   sci_ant.cmd -f localizedstringreconciler.xml switchtest
   -Dlanguage=xx -Dcountry=YY
   ```

   Here, xx and YY are the desired base language and country respectively, for example, fr and FR for French and France. This simulates the switch and reports any errors it finds, without making any changes to the database.

2. To perform the base language switch, run the LocalizedStringReconciler tool from <INSTALL_DIR>/bin in SWITCH mode as follows:

   For UNIX:

   ```
   sci_ant.sh -f localizedstringreconciler.xml switch
   -Dlanguage=xx -Dcountry=YY
   ```

   For Windows:

   ```
   sci_ant.cmd -f localizedstringreconciler.xml switch
   -Dlanguage=xx -Dcountry=YY
   ```

Here, xx and YY are the desired base language and country respectively, for example, fr and FR for French and France.

2.4.1.3 Full Export to Back Up Existing Localization Literals

To create a full export to back up the existing localization literals:

Run the LocalizedStringReconciler tool from <INSTALL_DIR>/bin in EXTRACT mode as follows:

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml extract
-Ddestdir=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml extract
-Ddestdir=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS
```

This exports the literals that are currently defined in the YFS_LOCALIZED_STRINGS table.

To extract the existing localized strings for a specific locale, use the optional parameters locale and country. The usage is as follows:

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml extract
-Ddestdir=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS -Dlanguage=fr -Dcountry=FR
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml extract
-Ddestdir=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS -Dlanguage=fr -Dcountry=FR
```

For example, if there are three locales that contain localized factory defaults in the database, one file is created for each locale in the following format:

```
<baselanguage>_<basecountry>_ycpdblocalizedstrings_<language>_
<country>_db.properties
```

Each file is saved within the extn folder within the respective locale directory. For instance, for the locale fr_FR, the file en_US_ycpdblocalizedstrings_fr_FR_db.properties is created and saved to the <INSTALL_DIR>/repository/factorysetup/complete_installation/XMLS/fr_FR directory.

To re-import the backed-up literals:

Run the LocalizedStringReconciler tool from <INSTALL_DIR>/bin in IMPORT mode with the file for which you have exported the literals as follows:

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml import
-Dsrc=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml import
-Dsrc=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS
```

2.4.1.4 Deleting Unused Localization Literals

To delete unused localization literals, run the LocalizedStringReconciler tool from <INSTALL_DIR>/bin in EXPORT mode as follows:

For UNIX:

```
sci_ant.sh -f localizedstringreconciler.xml export
-Ddestdir=<INSTALL_DIR>/repository/factorysetup/complete_
installation/XMLS -Ddelete=true
```

For Windows:

```
sci_ant.cmd -f localizedstringreconciler.xml export
-Ddestdir=<INSTALL_DIR>\repository\factorysetup\complete_
installation\XMLS -Ddelete=true
```

This deletes the unused literals when exporting the literals that are currently defined in the YFS_LOCALIZED_STRINGS table.

2.4.2 Master Data Localization for a Multi-Language Installation

Selling and Fulfillment Foundation enables the localization of master data such as items, categories, and assets. Localizing configuration data allows administrators to only configure localized literals for words and phrases. For example, if you localize the configuration data for a word Store, the localized literal for Store is displayed in every field that uses the word Store. However, master data localization allows you to specify the localized literal, for example, Super Computer, as an item's description. In this example, the localized literal Super Computer is displayed for the specified item's description. Administrators can use Business Center to localize master data in each locale and to modify master data localizations.

2.4.3 Data Localization in Entities for Locale-Specific Information

In addition to localized screen labels, localized values for the actual content (such as transaction data) must be stored and appropriate translation values displayed, according to the business usage or meanings across different locales.

To enable the display of localized data, the entity framework supports a child entity that stores different locale values for each column.

Columns which contain localized values must be marked as *Localized* in the parent entity. Such columns automatically become a part of the child entity.

Note: You must provide LANGUAGE, COUNTRY and VARIANT columns for the child entity. You must also define the primary key, indices and relationship between an entity and its localized description. Since the LANGUAGE, COUNTRY and VARIANT columns along with the primary key of the parent table are used to search for the localized values of a column, you must define a unique key for this combination.

To display locale-specific information:

1. A field called LocaleDescriptionForEntity is added to entity definition, which is used to store localized description in each child table. Each child table is identified by this field.

2. Set the value of this field to the name of the parent entity. This value should match the value of *Name* attribute in the parent entity.

   For example, if the entity definition to store localized values for a column in the parent table "PLT_QUALIFIED_TAG" is:

   ```
   <Entity Cacheable="true"
      Description="Stores qualifier information"
      EntityType="CONFIGURATION" AuditRequired="N"
      Module="ycp" Name="QualifiedTag" Prefix="PLT_"
      TableName="PLT_QUALIFIED_TAG" XMLName="QualifiedTag">
   <Attributes>
   ....
   ```

   The corresponding child entity definition should be:

   ```
   <Entity Description="Stores qualifier information"
      EntityType="CONFIGURATION" Name="Qualifier_Locale"
      Prefix="YFS_" TableName="YFS_QUALIFIER_LOCALE"
      XMLName="QualifierLocale"
      LocaleDescriptionForEntity="QualifiedTag">
      ...
   ```

   You can define child tables for Platform or lower stack entities and mark columns as localized. Each entity can have only one localized child entity.

3. Another field DisplayLocalizedFieldInLocale is added to the API layer. Pass this field to an API to indicate the locale from which the localized values must be displayed. The value of this field must be in the format, *language_country_variant (*fr_FR, en_US_CA):

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <QualifiedTag QualifiedTagId="" QualifiedTagKey="key1"
   DisplayLocalizedFieldInLocale=”fr_FR”/>
   ```

4. The **get** method for localized fields is modified to return the local values from the child tables, instead of the actual value.This method reads the locale set in the DisplayLocalizedFieldInLocale field and uses the language, country and variant information to obtain the corresponding values from the child table. For example, `getQualifiedTagDescription()`

   If there is no corresponding localized value, then the actual value from the parent entity is displayed.

5. To obtain the actual value (non-localized values) from the parent entity instead of the localized one, use the method `getNonLocalized`. For example, `getNonLocalizedQualifiedTagDescription()`

2.5 Directory Structure and File Names

The Selling and Fulfillment Foundation directory structure must be maintained based on the language and country localization. File names are limited to one locale and can only contain characters a through `z`, `0` (zero) through `9`, and the underscore (`_`).

Likewise, because many files are created using a transaction ID, this rule also applies to transaction names and transaction IDs. Similarly, you should adhere to these limitations when localizing the template file name for transaction events (`TransactionID.On_SUCCESS.xml`).

Selling and Fulfillment Foundation follows the hierarchy of lookups provided here:

1. When a certain locale is selected in the console, it searches for some files with that particular suffix specified in the locale. For example, if the locale `ja_JP` is chosen in the console, the system searches for files such as `ycpapibundle_ja_JP.properties`, `validation_ja_JP.properties`, `sapphire_ja_JP.css`, and so forth.
2. If these files are not found, the systems looks at the default locale of the operating system that the application server is running. If it is `en_US`, the system tries to look for the files mentioned in step 1, but with the suffix `en_US`, for example, `ycpapibundle_en_US.properties`.

3. Only if the first two steps fail are the default files, for example, `ycpapibundle.properties` used.

The method of defaulting to the standard files occurs for the files that can have locale suffix such as `alertmessages.js`, `scfoundationiconsbe.jar`, and so on.

2.6 User Interface Themes

The user interface theme files specify the screen colors and display fonts to be used. Display fonts are dependent on the languages that must be supported. Some fonts may not support all the languages. For example, Tahoma (the Selling and Fulfillment Foundation default font) does not support Japanese; for best results when localizing Selling and Fulfillment Foundation to Japanese, use the Microsoft Gothic font. When setting up a theme, choose a font that displays the specific language you require. For example, when setting up a Japanese locale, customize the theme to use a font that displays Japanese characters such as Hiragana.

> **Note:** If you use a font that is bigger than the Selling and Fulfillment Foundation default font (Tahoma), it may be necessary to customize the `<INSTALL_DIR>/repository/datatypes/datatypes.xml` file to increase the user interface size of data types that are used for input fields in the Application Console. In particular, the user interface size of the "Date" data type should be increased.

When choosing a font for a specific language, you can refer to Table 2–1 for the language's recommended font.

Table 2–1 Recommended Fonts for Languages

Language	Font
French	Tahoma
German	Tahoma
Japanese	MS UI Gothic
Spanish	Tahoma
Simplified Chinese	SimSun

Table 2–1 Recommended Fonts for Languages

Language	Font
Traditional Chinese	PMingLiu
Korean	Dotum

2.6.1 Localizing Themes

When localizing your Applications Manager user interface themes, you modify the theme-specific XML file. When localizing your Application Console user interface themes, you modify the theme-specific CSS file. For example, the following files must be localized for themes:

`<INSTALL_DIR>/repository/xapi/template/merged/resource/<theme>.xml` and `<INSTALL_DIR>/repository/eardata/platform/war/css/<theme>.css`

These files are localized by appending the language and country codes in the file name. For example, if you are using the sapphire theme in a French locale, localize the following files:

`<INSTALL_DIR>/repository/xapi/template/merged/resource/sapphire_fr_FR.xml`
`<INSTALL_DIR>/repository/eardata/platform/war/css/sapphire_fr_FR.css`

The following themes are distributed with Selling and Fulfillment Foundation:

- Earth (`<INSTALL_DIR>/repository/xapi/template/merged/resource/earth.xml` and `<INSTALL_DIR>/repository/eardata/platform/war/css/earth.css`)
- Jade (`<INSTALL_DIR>/repository/xapi/template/merged/resource/jade.xml` and `<INSTALL_DIR>/repository/eardata/platform/war/css/jade.css`)
- Sapphire (`<INSTALL_DIR>/repository/xapi/template/merged/resource/sapphire.xml` and `<INSTALL_DIR>/repository/eardata/platform/war/css/sapphire.css`)

To localize a theme:

1. Copy the <INSTALL_DIR>/repository/xapi/template/merged/resource/<theme>.xml file and save it as <INSTALL_DIR>/repository/xapi/template/merged/resource/<theme>_<language>_<country>.xml.
2. Copy the <INSTALL_DIR>/repository/eardata/platform/war/css/<theme>.css file and save it as <INSTALL_DIR>/repository/eardata/platform/war/css/<theme>_<language>_<country>.css.

   Edit the <INSTALL_DIR>/repository/eardata/platform/war/css/<theme>_<language>_<country>.css file to change the display font for the Application Console. In addition, the font name and size for the graph displayed in the Inventory Summary screen in the Inventory Console is configured in the <INSTALL_DIR>/repository/xapi/template/merged/resource/<theme>_<language>_<country>.xml file.

   For example, in the default sapphire.xml file, the graph font is configured as:

```
<!-- Font for Inventory Graphs(Axis Titles & Lables) -->
<Font Name="InvGraphFont" FontName="Tahoma" FontSize="12"/>
<!-- Font for Inventory Graphs -->
```

To localize double-byte languages such as Japanese, Sterling Commerce recommends that you edit the <INSTALL_DIR>/repository/xapi/template/merged/resource/<theme>_<language>_<country>.xml file to use either the MS UI Gothic or SimSun font as follows:

```
<!-- Font for Inventory Graphs(Axis Titles & Lables) -->
<Font Name="InvGraphFont" FontName="simsun" FontSize="12"/>
<!-- Font for Inventory Graphs -->
```

3. Rebuild the resources.jar by running the following command from the <INSTALL_DIR>/bin directory:

   ```
   /deployer.sh -t resourcejar
   ```

4. If you are using Oracle WebLogic or IBM WebSphere, rebuild the EAR.

> **Note:** If your application server is running on UNIX, the valid fonts that you can use are stored in the <JAVA_HOME>/jre/lib/font_properties.<file.encoding> file.

2.7 Literals

All Selling and Fulfillment Foundation components use a common resource bundle that contains literals displayed on the screens. Selling and Fulfillment Foundation enables you to customize and localize resource bundles as required.

In addition, literals used in customized screens have their own resource bundle and should also be considered during the localization process. For more information about localizing your Selling and Fulfillment Foundation customizations, see the *Selling and Fulfillment Foundation: Customizing Console JSP Interface for End User Guide*.

Note: Literals cannot be localized in:

- Condition Builder
- Order/Shipment Monitor
- Hard-coded literals in APIs

For a complete list of resource bundle literals, along with the screens on which those literals appear, see the `<DocumentationCD>/resource_mapping.htm` file.

2.7.1 Resource Bundles

Complete resource bundles are released in the `ycpapibundle.properties` and `yscpapibundle.properties` files with the localized versions of Selling and Fulfillment Foundation. Incremental updates are not provided. If you localize Selling and Fulfillment Foundation, it is your responsibility (or that of your third-party localization company) to compare and validate the differences between the resource bundles shipped with the product to those you have localized.

The resource bundles of Selling and Fulfillment Foundation are located in the `<INSTALL_DIR>/resources/ycpapibundle.properties` and `<INSTALL_DIR>/resources/yscpapibundle.properties` files.

To localize the resource bundles:

1. Copy the `<INSTALL_DIR>/resources/ycpapibundle.properties` file and save it as `<INSTALL_DIR>/resources/ycpapibundle_<language>_<country>.properties`.
2. Copy the `<INSTALL_DIR>/resources/yscpapibundle.properties` file and save it as `<INSTALL_DIR>/resources/yscpapibundle_<language>_<country>.properties`.
3. Each resource bundle contains a `<key>=<value>` pair, where key is the resource key and value is the literal displayed for the corresponding locale. Replace `<value>` with the translated value.
    - When localizing menus in the Applications Manager, by default, the accelerator key is the first character in a menu item. To

specify any other character as the accelerator key, insert an ampersand (&) just before that character.

- When localizing the console UI, be aware that changing the height or width of the text in the application may affect the layout of the screens. It may be necessary to customize certain screens to achieve optimal layout after the other localization steps are complete. For example, if the resource bundle contains translated literals that are lengthy, you may have to increase the width of the screen in order to accommodate the larger size of the translated literal.
- Some of the literals that have to be translated in the resource bundles contain data place holders. These data place holders indicate that the literal is displayed with one or more data values within the literal. For example, when the application displays the error message "Priority should be greater than X", where X could be any number. Because the location of "X" within the literal can be different for different languages, the resource bundle uses a place holder that can be placed anywhere in the literal during translation. The resource bundle entry looks like this:

  ```
  PRIORITY_ERROR_MESSAGE=Priority should be greater than {0}
  ```

  Notice how the "{0}" place holder indicates where the dynamic data value appears in the literal. This "{0}" can be placed anywhere in the literal, for example, the following two options are valid possibilities:

  ```
  PRIORITY_ERROR_MESSAGE=A number greater than {0} should be entered
  ```

  or

  ```
  PRIORITY_ERROR_MESSAGE={0}: Priority entered should be greater than this
  ```

  Placeholders give you the flexibility to translate the literal in any way the corresponding language dictates. Note that multiple place holders may appear in the literal as well, for example, {0}, {1}, {2}, and so on. Each place holder must exist somewhere in the corresponding translated literal.
- When using literals that contain data place holders, you cannot use single quotation marks. If a single quotation mark is used in conjunction with a place holder, the single quotation mark is not

displayed and the place holder is not replaced with its replacement value. In order to avoid this situation, enter two single quotes wherever a single quote is required. For example, the following statement is invalid:

```
PRIORITY_ERROR_MESSAGE=The primary organization’s name is {0}
```

However, the following statement is valid:

```
PRIORITY_ERROR_MESSAGE=The primary organization’’s name is {0}
```

- Files should be returned in native format with UTF-8 encoding.
- Properties should be returned in escaped Unicode format with UTF-8 encoding.

4. The default font used is Tahoma. Therefore if you want to display or type Unicode characters, you should localize the theme. This is done by changing the font to Unicode in the theme-specific XML files.
5. Save the modified file. If the file is in UTF-8 format, convert it to ASCII by running the `native2ascii` command as follows:

   ```
   native2ascii -encoding UTF-8 <source file> <target file>
   ```

   The file should be returned in the following format:

   ```
   <filename>_<2 letter code for language as given by ISO 639>_<2 letter code for territory as given by ISO 3166>.<file extension>
   ```

   For example, `ycpapibundle.properties` should be returned as `ycpapibundle_fr_FR.properties` and `yscpapibundle.properties` should be returned as `yscpapibundle_fr_FR.properties`.
6. If you are customizing Selling and Fulfillment Foundation, save the extended resource bundles as `<INSTALL_DIR>/resources/extn/extnbundle_<language>_<country>.properties`.

   For example, `ycpapibundle.properties` should be saved as `ycpapibundle_fr_FR.properties` and `yscpapibundle.properties` should be saved as `yscpapibundle_fr_FR.properties`.
7. For extended tag attributes, add the following bundle entry in `extnbundle.properties` for each extended tag attribute:

   ```
   Item_Tag_<TagName>=<Tag Name>
   ```

8. Create the resource jar using the `./deployer.sh -t resourcejar`.
9. If you are using Oracle WebLogic or IBM WebSphere, rebuild the EAR.

> **Note:** The Custom_Code_Prefix and Custom_Code_Postfix properties in the `ycpapibundles.properties` file are used to prefix or append identifying literals or extensions to your newly created custom transaction IDs, supply types, demand types, or document types. When you create a new transaction ID, supply type, demand type, or document type, the value specified for these properties is prefixed or appended to each of these types of literals when displayed in the user interface. The default value for the Custom_Code_Prefix is "" (blank) and the Custom_Code_Postfix is ".ex". You can change this value if it does not suit your requirements.

2.8 Validating Date, Time, and Number

Date and number validations are performed using JavaScript. By default, Selling and Fulfillment Foundation provides validation for the en_US locale (English for the United States).

2.8.1 Date and Time Validation

Dates can be stored in a standard format, but displayed according to the required format. If the date is entered on a screen, it must be verified. The date format is specified in the Applications Manager Locale Details screen. For more information, see the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

To localize date and time validation:

Save the `<INSTALL_DIR>/repository/eardata/platform/war/yfcscripts/Validation.js` file as `Validation_<language>_<country>.js` (with all translations in escaped Unicode format) and make modifications to the new file as indicated in the following steps.

1. Ensure that the date and time values match the entries specified in the Locale fields in the Applications Manager. For detailed information about the Locale fields and their suggested syntax, see the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
2. Change the [yfcDateFormat] (MM/dd/yyyy), [yfcTimeFormat] (HH:mm:ss) and [yfcDateTimeFormat] (MM/dd/yyyy HH:mm:ss) variables to contain the correct date and time format.

   These date and time formats are according to the United States English version. When you create a Validation.js file for another language, these formats change accordingly. The following table specifies the date and time formats.

Table 2–2 Date and Time Variable Formats

Date Element	Description
yyyy	Four-digit year, for example, 2009
MM	Two-digit month, for example, 05 to indicate May
dd	Two-digit day, for example, 01 to indicate the first day of the month
mm	Minutes
HH	Hours
ss	Seconds

3. Localize the pop-up messages and calendar formats as described in Section 2.10, "Calendar and Message Pop-Up Windows in the Console".

2.8.2 Number Validation

By default, Java displays the localized number format based on what is specified for <language>_<country> in the Applications Manager. For the Selling and Fulfillment Foundation UI to validate numeric values, the application must read the values as specified in the Validation_<language>_<country>.js file. This means that the validation JavaScript file must contain validation logic that is specific to the number format to be used for the validation for each locale.

To localize number validation:

1. Save the <INSTALL_DIR>/repository/eardata/platform/war/yfcscripts/Validation.js file as Validation_<language>_<country>.js (with all translations in escaped Unicode format).
2. Change the decimal separator and grouping separator, as required. The grouping separator indicates how to format numbers visually; it has no impact on the actual value of the number.

   For example, en_US uses a comma (",") and fr_FR uses a non-breaking space (" "). When specifying the separator, use the Unicode literal.

   For example, the non-breaking character for fr_FR will be specified as shown below:

   ```
   yfcGroupingSeparator = "\u00a0";
   ```

3. Localize the exception messages for invalid number format as described in Section 2.7, "Literals".
4. Run the resource deployer from ./deployer.sh -l info -t resourcejar.
5. If you are using Oracle WebLogic or IBM WebSphere, rebuild the EAR.

2.9 Templates

Among the templates used by Selling and Fulfillment Foundation, you can localize the e-mail and the exception alert templates.

2.9.1 E-Mail Templates

You can store e-mail templates using any character encoding format, but the encoding format must be set by configuring the yfs.email.template.encoding property in the <INSTALL_DIR>/properties/customer_overrides.properties file.

For additional information about overriding properties using the customer_overrides.properties file, see the *Selling and Fulfillment Foundation: Properties Guide*.

2.9.2 Exception Alert Templates

Exception alert templates enable you to provide additional text to the alerts raised. This enables you to make error messages more descriptive and easy to understand. They also provide a means of providing a hyperlink to the resolution screens from the Alert Console.

For example, for any alert created for an order, shipment, or load document type, a hyperlink is created and displayed in the "Created For" column in the Alert List screen. You localize the literals displayed in this column by translating them in the `DefaultListTemplate.xsl` file located in your `<INSTALL_DIR>/repository/xapi/template/merged/exception_console` directory.

You can store exception alert templates using any character encoding format, but the encoding format must be configured in the yfs.file.encoding property in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. If these properties are not explicitly set in the `<INSTALL_DIR>/properties/customer_overrides.properties` file, Selling and Fulfillment Foundation uses UTF-8 character encoding.

For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

2.10 Calendar and Message Pop-Up Windows in the Console

Calendar and message pop-up windows contain literals that must be localized.

Note: Only the literals in the calendar pop-up window can be localized. The order of the days cannot be localized. Therefore, the week always starts with Sunday.

To localize date formats for calendar popup windows in the Console:

1. Copy the <INSTALL_DIR>/repository/eardata/platform/war/common/alertmessages.js file and save it as <INSTALL_DIR>/repository/eardata/platform/war/common/alertmessages_<language>_<country>.js (with all translations in escaped Unicode format).
2. Copy the <INSTALL_DIR>/repository/eardatasmcfs/war/ysc/scripts/yscalertmessages.js file and save it as <INSTALL_DIR>/repository/eardatasmcfs/war/ysc/scripts/yscalertmessages_<language>_<country>.js (with all translations in escaped Unicode format).
3. Copy the <INSTALL_DIR>/repository/eardata/smcfs/war/ysc/scripts/yscalertmessages.js file and save it as <INSTALL_DIR>/repository/eardata/smcfs/war/ysc/scripts/yscalertmessages_<language>_<country>.js (with all translations in escaped Unicode format).
4. Translate the following date-related literals:
   - monthArray - Contains the series of literals to be used for months of the year
   - weekdayList - Contains the series of literals to be used when displaying the literals for days of the week
   - weekdayArray - Contains the series of literals to be used when displaying the shortened literals for days of the week
   - todayString - Displays the word "Today"
5. If you are using Oracle WebLogic or IBM WebSphere, rebuild the EAR.

To localize exception messages:

1. Copy the <INSTALL_DIR>/repository/eardata/platform/war/common/alertmessages.js file and save it as <INSTALL_DIR>/repository/eardata/platform/war/common/alertmessages_<language>_<country>.js, (with all translations in escaped Unicode format).

2. Copy the <INSTALL_DIR>/repository/eardata/smcfswar/ysc/scripts/yscalertmessages.js file and save it as <INSTALL_DIR>/repository/eardata/smcfswar/ysc/scripts/yscalertmessages_<language>_<country>.js (with all translations in escaped Unicode format).
3. Copy the <INSTALL_DIR>/repository/eardata/smcfs/war/ysc/scripts/yscalertmessages.js file and save it as <INSTALL_DIR>/repository/eardata/smcfs/war/ysc/scripts/yscalertmessages_<language>_<country>.js (in UTF-8 encoding format).
4. Edit your exception messages in the files, as required.

2.11 Images

Images and icons are stored in JAR files. A separate JAR file can be used for each locale. Both the Applications Manager and Application Console use their own mechanisms for reading images.

2.11.1 Application Consoles Images

To localize images used in the Application Console:

1. Copy the <INSTALL_DIR>/repository/eardata/war/yfscommon/scfoundationiconsbe.jar file to a new <INSTALL_DIR>/repository/eardata/war/yfscommon/scfoundationiconsbe_<language>_<country>.jar file.
2. Within your custom JAR file, add or remove images, as required.
3. If you are customizing and localizing Selling and Fulfillment Foundation, your custom images are read from the <INSTALL_DIR>/repository/eardata/extn/war/icons/scfoundationiconsbe.jar file. Therefore, you must copy the images from the <INSTALL_DIR>/repository/eardata/extn/war/icons/scfoundationiconsbe.jar file and make the necessary locale-specific changes in the images.

Save the changed images in the <INSTALL_DIR>/repository/eardata/extn/war/icons/scfoundationiconsbe_<language>_<country>.jar file.

Note: The warning message that is displayed when you click the Selling and Fulfillment Foundation icon in the top right-hand corner of the Application Console user interface, is included as text in an image file (about_box_back.gif). You can localize this message by following the instructions in this section.

2.11.2 Applications Manager Images

To localize the images used in the Applications Manager:

1. Copy the images from the <INSTALL_DIR>/repository/eardata/war/yfscommon/scfoundationicons.jar file and make the necessary locale-specific changes to the images.
2. Save the changed images in the <INSTALL_DIR>/repository/eardata/war/yfscommon/scfoundationicons_<language>_<country>.jar file.

> **Note:** The warning message that displays at the bottom of the Applications Manager **About box** when you select **Help > About from the Applications Manager** menu is included as text in an image file (about.gif). You can localize this message by following the instructions in this section.

2.12 Localizing the Greex File

A Greex file, also known as advanced XML file, contains the advanced XML condition or Greex Rule that is defined by a user. By localizing the Greex file, you can localize an advanced XML condition or Greex Rule.

To localize a Greex file:

1. Create the BundleResolver class and implement the following methods within the class:

- getString(String key) method and return the localized strings.

  For example, to localize a Greex file using properties files:

```
public class MyBundleResolver implements BundleResolver
{
   Properties prop = new Properties();
     public MyBundleResolver()
     {
       //read and initialize the property file
     }
     public  String getString(String key)
     {
         Return prop.getProperty(key);
     }
 }
```

2. Register the BundleResolver class with the GreexContext using the registerBundle() method, for example:

```
public class MyApp
{
  GreexContext  ctx = new GreexContext();
  ctx.registerBundle(new MyBundleResolver())
}
```

2.13 Localizing the Context-Sensitive Help

During installation, the Context-Sensitive Help is placed under the `<INSTALL_DIR>/xapidocs/online_help` directory.

To localize the Context-Sensitive Help of the Application:

1. Create a directory for the desired locale. For example, if you are translating the Context-Sensitive Help to German, create the following directory for files created in the `online_help` directory:

   `<INSTALL_DIR>/xapidocs/online_help/de_DE/`

2. Copy the files in the `<INSTALL_DIR>/xapidocs/online_help/` to the locale-specific directory.

3. Translate the files in the locale-specific directory.

4. Add `yfs.onlinehelp.path.overrideforlocale.<locale>` to the `customer_overrides.properties` file.

   For translating to the German locale, set the `yfs.onlinehelp.path.overrideforlocale.de_DE` property to `/smcfsdocs/yfscommon/online_help/de_DE`.

   For additional information about overriding properties using the `customer_overrides.properties` file, see the *Selling and Fulfillment Foundation: Properties Guide*.

5. Re-build and deploy the Selling and Fulfillment Foundation EAR. For more information on how to build an EAR, see the *Selling and Fulfillment Foundation: Installation Guide*.

If you are installing both Selling and Fulfillment Foundation and the Language Pack together, a one-time creation and deployment of the EAR file is sufficient. If you have already deployed your application and are planning to install the Language Pack later, you must re-create and redeploy the EAR file.

For more information about creating and deploying the EAR file for your chosen application server, see the *Selling and Fulfillment Foundation: Installation Guide*.

Note: If you do not want to translate the Context-Sensitive Help, you can use the default US English files, but the non-translated files must be copied to the locale-specific directory, as described in Step 2. If this is not done, a user who is logged in under a non-US locale will receive an "Error 404--Not Found" message when accessing the Context-Sensitive Help.

2.14 Localizing the Synonyms for Catalog Search

Selling and Fulfillment Foundation allows you to configure synonyms for catalog search. Synonyms enable a catalog search to return an expanded list of items based on related search terms. For example, if notebook is configured as a synonym for laptop, and laptop is entered as a search term, the search results return items identified as notebooks as well as laptops.

By default, Application does not provide synonyms for search terms. However, you can configure synonyms for each of the locales your catalog search supports. For example, if your catalog search supports the US-English and French locales, you can configure US-English and French synonyms.

To define synonyms for a locale:

1. In the extended XML configuration file for item search, specify a path for the synonym file in the Locale element. For information about customizing item search, see the *Extending the Database Guide*.
2. In the `Properties` directory, open the synonym file that corresponds to the file specified in the Locale element of the extended XML configuration file for item search.
3. Specify related terms using the following format:

   ```
   term1=synonym1, synonym2, synonym3
   term2=synonym4, synonym5
   ```

   For example, to configure coffee maker as a synonym for coffee machine, specify:

   ```
   coffee machine=coffee maker
   ```

> **NOTE:** The mapping of related terms for synonyms is not reciprocal. In the example provided previously, coffee maker is a synonym for coffee machine, but coffee machine is not specified as a synonym for coffee maker. If a customer enters coffee machine as a search term, the search returns results pertaining to both coffee machine and coffee maker. However, if a customer enters only coffee maker as the search criteria, the search returns only coffee maker.

3

Localizing the Rich Client Platform Application

This chapter provides the steps involved in localizing the Rich Client Platform application for the languages that are not yet provided in the Application and the languages that are supported by Selling and Fulfillment Foundation out of the box.

3.1 Prerequisites to Localizing the Rich Client Platform Application

Before you begin to localize a Rich Client Platform application, read the information provided in this section.

International Standards Organization Codes

Rich Client Platform uses locale-related codes as specified by the International Standards Organization (ISO). Throughout this guide, locale is represented by <language>_<country>. Locale comprises of <language>, a 2-character lower-case ISO-639 code, and <country>, a 2-character upper-case ISO-3166 code.

Note: For complete steps to localize the Rich Client Platform Application, see the Implementation Guide specific to your PCA.

3.2 User Interface Themes

The user interface theme files specify screen colors, display fonts, and images to use. Display fonts are dependent on the languages that are supported. Some fonts may not support all languages. When setting up a theme, select a font that displays the specific language you require or select an image that is displayed for a particular locale. For example, when setting up a Japanese locale, customize the theme to use a font that displays Japanese characters, such as Hiragana.

3.2.1 Localizing the Rich Client Platform Application Themes

When localizing your Rich Client Platform user interface themes, localize the Rich Client Platform-specific theme files and the Rich Client Platform application-specific theme files.

The Rich Client Platform contains `com.yantra.yfc.rcp.common_<theme_name>.ythm` theme file, which is located in the `com.yantra.yfc.rcp.common_<version>` directory that is available in the `<INSTALL_DIR>/repository/rcp/rcpclient/com.yantra.yfc.rcp_<version>.zip` file, once extracted.

The Rich Client Platform contains `<Plug-in_ id>_<theme_name>.ythm` theme file, which is located in the `plugins/com.yantra.pca.ycd.rcp_<version>` directory within the `<INSTALL_DIR>/repository/rcp/rcpclient/com20.zip` file.

The theme file is localized by appending the language and country codes in the file name. For example, if you are localizing the sapphire theme file in a French locale of the Rich Client Platform, modify the `com.yantra.yfc.rcp.common_sapphire.ythm` file as:

`com.yantra.yfc.rcp.common_sapphire_fr_FR.ythm`

To localize the sapphire theme file in a French locale of the Rich Client Platform application, modify the `<Plug-in_id>_sapphire.ythm` file as:

`<Plug-in_id>_sapphire_fr_FR.ythm`

For example, if you are localizing the sapphire theme file in a French locale of the Sterling Call Center and Sterling Store PCA, modify the com.yantra.pca.ycd.rcp_sapphire.ythm file as:

com.yantra.pca.ycd.rcp_sapphire_fr_FR.ythm

3.2.1.1 Types of themes

There are two types of themes, Fonts and Images.

Fonts

Theme entries for the Font theme are specified in the Font, ForegroundColor, and BackgroundColor elements under the ThemeEntry element. There are three types of theme entries for the Font theme:

- Font—The Font element contains data pertaining to the font's height, name, and style. Table 3–1 describes the attributes of the Font element.

Table 3–1 Font Element Attribute List

Attribute	Description
Height	Enter the height of the font.
Name	Enter the font name, for example, Tahoma, Arial, and so on.
Style	Enter the font style, for example, NORMAL, BOLD, and so on.

- ForegroundColor and BackgroundColor—The ForegroundColor element describes the color of the text that is displayed. The BackgroundColor element describes the background on which the text is displayed. Available attributes for both elements are red, blue, and green. These standard RGB values must be entered in the decimal color code (0-255).

Images

The theme entries for the Image theme are specified in the Image element under the ThemeEntry element. In the Image element, specify the path for the locale-specific image to be used in the theme in the Path attribute.

To store the locale-specific images:

1. Create an icons folder in the <INSTALL_DIR>/repository/rcp/extn directory.

2. Put all the locale specific images in this newly created <INSTALL_DIR>/repository/rcp/extn/icons folder.

Following is a sample data from the <Plug-in_ id>_<theme_name>.ythm file:

```
<?xml version="1.0" encoding="UTF-8"?>
<Theme id="sapphire">
  <ThemeEntry Name="Label">
     <Font Height="9" Name="Tahoma" Style="NORMAL"/>
     <ForegroundColor Blue="0" Green="0" Red="0"/>
     <BackgroundColor Blue="255" Green="255" Red="255"/>
  </ThemeEntry>
  <ThemeEntry Name="CComboEditor">
     <Font Height="8" Name="Tahoma" Style="NORMAL"/>
     <BackgroundColor Blue="255" Green="255" Red="255"/>
     <ForegroundColor Blue="0" Green="0" Red="0"/>
  </ThemeEntry>
  <ThemeEntry Name="Text">
     <Font Height="8" Name="Tahoma" Style="NORMAL"/>
     <ForegroundColor Blue="0" Green="0" Red="0"/>
     <BackgroundColor Blue="255" Green="255" Red="255"/>
  </ThemeEntry>
 <ThemeEntry Name="Composite">
     <Font Height="10" Name="Tahoma" Style="NORMAL"/>
     <BackgroundColor Blue="255" Green="255" Red="255"/>
     <ForegroundColor Blue="198" Green="146" Red="140"/>
 </ThemeEntry>
 <ThemeEntry Name="DateLookup">
     <Image Path="/icons/calendar.gif"/>
 </ThemeEntry>
 <ThemeEntry Name="HeaderTriangle">
     <Image Path="/icons/header_triangle.jpg"/>
 </ThemeEntry>
```

3.2.1.2 Localizing a Theme

To localize the client application theme files, perform the following tasks:

1. Navigate to the <INSTALL_DIR>/repository/rcp/rcpclient/ directory.
2. Unzip the com.yantra.yfc.rcp_<version>.zip file. All Rich Client Platform-specific theme files are located in the <unzip_dir>/com.yantra.yfc.rcp.common_<version> directory.

Here, <unzip_dir> refers to the directory in which you have unzipped the file.

The theme files are in the following format:

com.yantra.yfc.rcp.common_<theme_name>.ythm

Here, <theme_name> refers to the name of a particular theme. By default, Rich Client Platform provides three different theme files named sapphire, jade, and earth.

3. Copy the com.yantra.yfc.rcp.common_<theme_name>.ythm file to the resources directory under the extensions folder that you created. For more information about creating an extensions folder, see the *Selling and Fulfillment Foundation: Installation Guide*.
4. Navigate to the <INSTALL_DIR>/repository/rcp/rcpclient/ directory.
5. Unzip the <PCA_APPLICATION_ID_VERSION>.zip file and navigate to the following directory:

   <unzip_dir>/plugins/<Plug-in_id>_<version>

   Here, <unzip_dir> refers to the directory in which you have unzipped the <PCA_APPLICATION_ID_VERSION>.zip file.

   In the <Plug-in_id>_<version> directory, there is a <PCA_APPLICATION_ID_VERSION>.jar file. The theme file is in the root directory of this JAR file and is called <Plug-in_id>_<theme_name>.ythm.
6. Copy the <Plug-in_id>_<theme_name>.ythm file to the resources directory under the extensions folder that you created.

   For example, if you are localizing a theme file of the Sterling Call Center and Sterling Store PCA, unzip the <INSTALL_DIR>/repository/rcp/rcpclient/com20.zip file.
7. Navigate to the <unzip_dir>/plugins/com.yantra.pca.ycd.rcp_<version> directory.

   Here, <unzip_dir> refers to the directory in which you have unzipped the com20.zip file.

   In the com.yantra.pca.ycd.rcp_<version> directory, there is a com20.jar file. The theme file is in the root directory of this JAR file and is called com.yantra.pca.ycd_<theme>.ythm.

8. Copy the com.yantra.pca.ycd_<theme>.ythm file to the resources directory under the extensions folder that you created.
9. Rename the <Plug-in_ id>_<theme_name>.ythm files as <Plug-in_ id>_<theme_name>_<locale_name>.ythm and the com.yantra.yfc.rcp.common_<theme_name>.ythm files as com.yantra.yfc.rcp.common_<theme_name>_<locale_name>.ythm.

   The <locale_name>=lang_cc, where lang refers to the language code and cc refers to the country code. For example, com.yantra.pca.ycd_ sapphire_en_US.ythm.
10. Modify the theme entries in the theme file based on the user's locale. For more information about theme entries in the theme file, see Section 3.2.1.1, "Types of themes".
11. Create a new icons folder and store all the images or icons that you want to localize in this folder. Now, copy the icons folder to the <RCP_EXTN_FOLDER>/resources directory.
12. If you are using Oracle WebLogic or IBM WebSphere, rebuild the EAR.

Note: For information on localizing theme files for PCAs, see the Implementation Guide specific to your PCA.

3.3 Rich Client Platform Application Literals

All Rich Client Platform applications use a resource bundle that contains literals or text displayed on the screens. Rich Client Platform enables you to customize and localize resource bundles based on a user's locale.

In addition, literals used in customized screens have their own resource bundle and should also be considered during the localization process.

3.3.1 Resource Bundles

All Rich Client Platform application plug-ins contain bundle files named as <Plug-in_id>_<name>.properties. The bundle file contains <key>-<value> pairs, and specifies resources such as control text, string, and so on that have to be localized in the <Key> and the localized string or text in <value>.

Following is an example of <key>=<value>:

my_name=Rich Client Platform Application

The resource bundles of a Rich Client Platform application are located in its zip file.

For example, the resource bundle for the Sterling Call Center and Sterling Store PCA is located in the <INSTALL_DIR>/repository/rcp/rcpclient/com20.zip file.

Unzip the com20.zip file, and navigate to the following directory:

<unzip_dir>/plugins/com.yantra.pca.ycd.rcp_<version>

Here, <unzip_dir> refers to the directory in which you have unzipped the com20.zip file.

Within this directory, there is a com20.jar file. The bundle files are in the root directory of this JAR file and are called com.yantra.pca.ycd_bundle.properties and com.yantra.pca.ycd_Messages_bundle.properties.

To localize the resource bundles:

1. Copy the <Plug-in_id>_<name>.properties file from the Rich Client Platform application plug-in's directory of the application that you want to localize, and also from the Rich Client Platform plug-in directory to the resources directory under the extensions folder that you created.

   Note: Because every Rich Client Platform application is dependent on the Rich Client Platform plug-in, whenever you want to localize a Rich Client Platform application, you must modify the <Plug-in_id>_<name>.properties file of both the Rich Client Platform plug-in and the Rich Client Platform application plug-in directories. The Rich Client Platform plug-in directory is located in the <INSTALL_DIR>/repository/rcp/rcpclient/com.yantra.yfc.rcp_<version>.zip file.

   To localize the Sterling Call Center and Sterling Store PCA bundle file, follow the instructions provided in the example below.

2. For example, if you are localizing the bundle file of the Sterling Call Center and Sterling Store PCA, the <INSTALL_DIR>/repository/rcp/rcpclient/com20.zip file will contain the Sterling COM PCA plug-in directory called com.yantra.pca.ycd.rcp_<version> directory.
3. Unzip the com20.zip file, and navigate to the following directory:

   <unzip_dir>/plugins/com.yantra.pca.ycd.rcp_<version>

   Here, <unzip_dir> refers to the directory in which you have unzipped the com20.zip file.

   Within this directory, there is a com20.jar file. The bundle file is in the root directory of this JAR file and is called com.yantra.pca.ycd_<name>.properties.

   Therefore, copy the <Plug-in_id>_<name>.properties file from both the Sterling Call Center and Sterling Store PCA plug-in directory as well as from the Rich Client Platform plug-in directory to the resources directory under the extensions folder that you created. For more information about creating the extensions folder, refer to the *Selling and Fulfillment Foundation: Installation Guide*.
4. Rename the <Plug-in_id>_<name>.properties file as <Plug-in_id>_<name>_<locale_name>.properties. The <locale_name>=lang_cc, where lang refers to language code and cc refers to the country code, for example, com.yantra.pca.ycd_bundle_en_US.properties.
5. Each resource bundle contains a <key>=<value> pair where key is the resource key and value is the literal displayed for the corresponding locale. Replace the <value> with the translated value.

Note: By default, Rich Client Platform localizes:

- Text on Labels
- Table Column names
- Descriptions in Combo Boxes
- Text on Buttons
- Tab Folder items
- Groups names

Rich Client Platform does not localize the text in the text boxes and the keys used for identification, such as ItemId or a resource key.

Following are the sample bundle entries from the `*.properties` file:

```
Credit_Card_#=Credit Card
View_Details=View &Details
Customer_Address=Customer Address
Save=&Save
Ship_To_Address=Ship To Address
Address=Address
Close=&Close
```

Here, entries on the left represent the resource key and entries on the right represent the translated value that is displayed for each control, text, or string, based on the user's locale.

6. If you want to get the localized value for any key, use the following method:

```
YRCPlatformUI.getString(String bundleKey);
```

It returns the localized string as the output.

Note: For information on localizing resource bundles for PCAs, see the Implementation Guide specific to your PCA.

3.3.1.1 Eclipse Resource Bundles

To localize the Eclipse platform resource bundles:

1. Modify the Rich Client Platform application's `*.ini` file to provide the appropriate program arguments to use the language pack. You can find the `*.ini` file for the Rich Client Platform application in the `<INSTALL_DIR>/bin/rcpdrop/<OPERATING_SYSTEM>/<PCA_DIR>/` directory.

   Here, `<INSTALL_DIR>` refers to the directory in which you have installed Selling and Fulfillment Foundation.

   For example, if you want to run the Sterling COM PCA in debug mode, edit the `<INSTALL_DIR>/bin/rcpdrop/<OPERATING_SYSTEM>/com/com.ini` file.

2. In the `*.ini` file, add one of the following arguments:

   - Program arguments

     ```
     -nl
     <locale_code>
     ```

   - VM arguments

     ```
     -Duser.language=<language_code>
     ```

Note: You must enter the program arguments before the VM arguments. Also, ensure that the program arguments are separated by a line break.

A

Localizable XML Attributes

Table A–1, "Localizable Factory Setup XML Attributes" enables you to determine the XML file name associated with the corresponding XML attributes. This table lists the factory default XML attributes that can be localized along with the associated database table name. The full name of the XML file can be derived from the database table name using the procedure described here.

To derive a list of XML files:

1. Find the database table that corresponds to the XML attribute you want to translate.
2. Using the database table name, append ".xml" to it and prepend the appropriate prefix for the module to which it belongs. Module prefixes are as follows:
   - BI—Business Intelligence
   - INV—Global Inventory Visibility
   - OMD—Order Management, Supply Collaboration, and Reverse Logistics
   - OMP—Order Management Platform
   - OMR—Order Management Returns
   - OMS—Order Management Services
   - RDT—Rapid Deployment
   - REF—Reference implementation
   - VAS—Value Added Services
   - WMS—Warehouse Management Services

- YCM—Catalog Management
- YCP/PLT—Platform
- YCS—Carrier Service
- YDM—Delivery Management
- YSC—Supply Chain

For example, the `Actionname` attribute corresponds with the YFS_ACTION database table, which uses the `YCP_YFS_ACTION.xml` file.

Table A–1 Localizable Factory Setup XML Attributes

Database Table Name	XML Attribute Name
SI_VERSION	ProductLabel
YFS_ACTION	Actionname GroupId
YFS_ACTIVITY	Description
YFS_ACTIVITY_CONSTRAINT	ConstraintDescription
YFS_ADAPTER	AdapterDescription
YFS_ADJUSTMENT_REASON	Description
YFS_ALLOCATION_RULE	Description
YFS_ANSWER_OPTION	Description
YFS_ATP_RULES	AtpRuleName
YFS_ATTR_ALLOWED_VALUE	ShortDescription LongDescription
YFS_ATTRIBUTE	ShortDescription LongDescription
YFS_ATTRIBUTE_DOMAIN	ShortDescription LongDescription
YFS_ATTRIBUTE_GROUP	ShortDescription LongDescription
YFS_BARCODE_TRANSLATION	Description

Table A–1 Localizable Factory Setup XML Attributes

Database Table Name	XML Attribute Name
YFS_BASE_ACTIVITY_GROUP	ActivityGroupName
YFS_BASE_DOCUMENT_ PARAMS	Description
YFS_BASE_EVENT	EventName
YFS_BASE_PACK_DO_NOT_MIX	FieldNameDesc
YFS_BASE_PROCESS_TYPE	ProcessTypeName
YFS_BASE_PURGE_CRITERIA	PurgeCodeDescription
YFS_BASE_SHIP_CONSTRAINTS	FieldNameDesc
YFS_BASE_TRANSACTION	BaseTranname
YFS_BUSINESS_DOCUMENT	DocumentDescription DocumentName
YFS_CALENDAR	CalendarDescription
YFS_CARRIER_SERVICE	CarrierServiceDesc
YFS_CATEGORY	ShortDescription Description
YFS_CATEGORY_DOMAIN	ShortDescription Description
YFS_CHARGE_CATEGORY	Description
YFS_CHARGE_NAME	Description
YFS_CLASSIFICATION_ PURPOSE	ClassificationPurposeDesc
YFS_COMMON_CODE	CodeLongDescription CodeShortDescription **Note**: For entries where CODE_TYPE starts with CODE_TYPE_LIST, change the CODE_DESCRIPTION property in the `ycpapibundle.properties` and `yscpapibundle.properties` files.

Table A–1 Localizable Factory Setup XML Attributes

Database Table Name	XML Attribute Name
YFS_CONDITION	ConditionName
YFS_CONFIG_VERSION_LABEL	Description
YFS_COUNT_PROGRAM	CountProgramName
YFS_COUNT_PROGRAM_COND	Description
YFS_COUNT_STRATEGY	Description
YFS_CURRENCY	CurrencyDescription
YFS_DATE_TYPE	Description
YFS_DEPARTMENT	DepartmentName
YFS_DEVICE_SUB_TYPE	Description
YFS_DEVICE_TYPE	Description
YFS_DOCUMENT_FORMAT	FormatDescription
YFS_DOCUMENT_PARAMS	Description
YFS_ENTERPRISE	Enterprisename
YFS_EQUIPMENT_TYPE	Description
YFS_ERROR_CAUSE_ACTION	Cause Action
YFS_ERROR_CODE	ErrorMessage
YFS_EVENT	EventName ExceptionType
YFS_FLOW	FlowName FlowGroupName
YFS_EXCEPTION_TYPE	ExceptionTypeDescription
YFS_EXECUTION_EXCEPTION	ExceptionShortDescription ExceptionLongDescription
YFS_FREIGHT_TERMS	Description ShortDescription

Table A–1 Localizable Factory Setup XML Attributes

Database Table Name	XML Attribute Name
YFS_HM_GROUPS	GroupName
YFS_HOLD_TYPE	HoldTypeDescription
YFS_IBA_SEQUENCE_HDR	Description
YFS_INVENTORY_DEMAND_ TYPE	Description
YFS_INVENTORY_STATUS	Description
YFS_INVENTORY_SUPPLY_TYPE	Description
YFS_ITEM_ATTR	ItemAttributeDescription
YFS_ITEM_UOM_MASTER	Description
YFS_LINE_RELATIONSHIP_TYPE	RelationshipShortDescription RelationshipLongDescription
YFS_LOCALE	LocaleDescription
YFS_LOCATION_SIZE	Description
YFS_MONITOR_TYPE	Field1Name, Field2Name, Field3Name, Field4Name, Field5Name, Field6Name, Field7Name, Field8Name, Field9Name, Field10Name
YFS_MONITORING_ CONSOLIDATION	Description
YFS_NEGOTIATION_RULE	NegotiationRuleId
YFS_NODE_TYPE	NodeTypeDescription
YFS_ORDER_LINE_TYPE	LineTypeDesc
YFS_ORDER_TAG_DETMN	Description
YFS_ORGANIZATION	OrganizationName **Note**: OrganizationCode and OrganizationKey cannot be localized.
YFS_PAYMENT_TYPE	PaymentTypeDescription
YFS_PICK_STRATEGY	Description

Table A–1 Localizable Factory Setup XML Attributes

Database Table Name	XML Attribute Name
YFS_PIPELINE	PipelineDescription
YFS_PLA_ACTIVITY_DETER	Description
YFS_PLA_ZONE_SET	Description
YFS_PRINT_DOCUMENT	PrintDocumentDescription
YFS_PROCESS_TASK_TYPE	ProcessTaskTypeDesc
YFS_PROCESS_TYPE	ProcessTypeName Description
YFS_PRODUCTIVITY_TYPE	Description ProductivityType
YFS_PURGE_CRITERIA	PurgeCodeDescription ErrFileName LogFileName
YFS_QUESTION	QuestionText
YFS_QUEUE	QueueDescription **Note**: QueueKey and QueueId cannot be localized.
YFS_REGION	RegionDescription
YFS_REGION_LEVEL	AddressFieldAlias
YFS_REGION_MATCH_PREF	AddressFieldAlias
YFS_RETRIEVAL_STRATEGY	Description
YFS_RETURN_DISPOSITION	Description
YFS_ROLE	RoleDescription **Note**: RoleID and RoleKey cannot be localized.
YFS_ROLE_DOCUMENT	DocumentName DocumentDescription
YFS_RULES	RuleSetFieldDescription

Table A–1 Localizable Factory Setup XML Attributes

Database Table Name	XML Attribute Name
YFS_SCAC	ScacDesc **Note**: Scac and ScacKey cannot be localized.
YFS_SCAC_AND_SERVICE	ScacAndService ScacAndServiceDesc
YFS_SERVICE_SLOT	ServiceSlotDesc
YFS_SERVICE_SLOT_GROUP	ServiceSlotGroupDesc
YFS_SERVICE_TYPE	Description
YFS_SHIP_NODE	Description
YFS_SHIPMENT_GROUP	Description
YFS_SPECIAL_SERVICES	SpecialServicesDescription
YFS_STATUS	Description StatusName
YFS_STATUS_MODIFICATION_TYPE	ModificationLevelScreenName ModificationTypeScreenName
YFS_TASK_TYPE	TaskTypeName
YFS_TRANSACTION	Tranname
YFS_TRANSACTION_DEPENDENCY	TransactionDependencyName
YFS_TRANSACTION_DEPENDENCY_GRP	GroupID GroupDescription
YFS_UOM	UomDescription **Note**: If `YFS_UOM.xml` is changed, the `YFS_UOM_CONVERSION.xml` should also have corresponding changes.
YFS_USER	Username
YFS_USER_EXIT_IMPL	ImplementationNotes
YFS_USER_GROUP	UsergroupName

Table A–1 Localizable Factory Setup XML Attributes

Database Table Name	XML Attribute Name
YFS_VOICE_APPLICATION	Description
YFS_VOICE_APPLICATION_ INSTANCE	Description
YFS_VOICE_WORK_FLOW	Description
YFS_WAVE_SIZE_CONSTRAINT	Description
YFS_ZONE	Description
YPM_PRICELIST_HDR	Description
YPM_PRICING_RULE	Description

Index

C

D

E

F

I

J

L

M

N

R

T

U