

Sterling Multi-Channel Fulfillment Solution

Installation Guide

Release 8.0

January 2008



Copyright Notice

Copyright © 1999 - 2008

Sterling Commerce, Inc. ALL RIGHTS RESERVED

STERLING COMMERCE SOFTWARE

TRADE SECRET NOTICE

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

This documentation, the Sterling Commerce Software it describes, and the information and know-how they contain constitute the proprietary, confidential and valuable trade secret information of Sterling Commerce, Inc., its affiliated companies or its or their licensors, and may not be used for any unauthorized purpose, or disclosed to others without the prior written permission of the applicable Sterling Commerce entity. This documentation and the Sterling Commerce Software that it describes have been provided pursuant to a license agreement that contains prohibitions against and/or restrictions on their copying, modification and use. Duplication, in whole or in part, if and when permitted, shall bear this notice and the Sterling Commerce, Inc. copyright notice.

U.S. GOVERNMENT RESTRICTED RIGHTS. This documentation and the Sterling Commerce Software it describes are "commercial items" as defined in 48 C.F.R. 2.101. As and when provided to any agency or instrumentality of the U.S. Government or to a U.S. Government prime contractor or a subcontractor at any tier ("Government Licensee"), the terms and conditions of the customary Sterling Commerce commercial license agreement are imposed on Government Licensees per 48 C.F.R. 12.212 or 227.7202 through 227.7202-4, as applicable, or through 48 C.F.R. 52.244-6.

These terms of use shall be governed by the laws of the State of Ohio, USA, without regard to its conflict of laws provisions. If you are accessing the Sterling Commerce Software under an executed agreement, then nothing in these terms and conditions supersedes or modifies the executed agreement

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

Copyright © 1999 - 2008

THIRD PARTY SOFTWARE AND OTHER MATERIAL

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 © 1999-2005 The Apache Software Foundation. Copyright 1999-2007 Erik Arvidson and Emil A. Eklund. Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin. Copyright © 2003 Infragistics, Inc. Copyright © 2001 LOOX Software, Inc. Copyright 2002-2004 © MetaStuff, Ltd. Copyright (C) Microsoft Corp. 1981-1998. Copyright © 1999-2005 Northwoods Software Corporation. Copyright © 2001 Peter Belesis. Copyright © 1995 - 1998 Purple Technology, Inc. Copyright © 2005 Sabre Airline Solutions. Copyright (c) 2006-2007 Sam Stephenson. Copyright © 2004 SoftComplex, Inc. Copyright © 2000-2004 Sun Microsystems, Inc. Copyright © 2001 VisualSoft Technologies Limited. Copyright © 1994 - 2007 World Wide Web Consortium (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). Copyright © 1998-2000 World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). Copyright © 2001 Zero G Software, Inc. All rights reserved by all listed parties.

The Sterling Commerce Software is distributed on the same storage media as certain Third Party Software covered by the following copyrights: Copyright © 1999-2006 The Apache Software Foundation. Copyright (c) 2001-2003 Ant-Contrib project. Copyright © 1998-2007 Bela Ban. Copyright © 2005 Eclipse Foundation. Copyright © 2002-2006 Julian Hyde and others. Copyright © 1997 ICE Engineering, Inc./Timothy Gerard Endres. Copyright © 1987-2006 ILOG, Inc. Copyright © 2000-2006 Infragistics. Copyright © 2002-2005 JBoss, Inc. Copyright LuMriX.net GmbH, Switzerland. Copyright © 1995-2002 MySQL AB. Copyright © 1999-2002 JBoss.org. Copyright Raghu K, 2003. Copyright © 2004 David Schweinsberg. Copyright © 2005-2006 Darren L. Spurgeon. Copyright © S.E. Morris (FISH) 2003-04. Copyright © 2006 VisualSoft Technologies. All rights reserved by all listed parties.

The Sterling Commerce Software is designed to be compatible with or implement a variety of standards issued by Third Party Licensors. The Sterling Commerce Software and related documentation may include copyrightable material of such Third Party Licensors, such as: Copyright © 2006 APCA [Australian Payments Clearing Association Limited]. Copyright © European Central Bank, Frankfurt am Main, Germany. Copyright © 2006 Fix Protocol Limited. Copyright Year 2002-2006 IFX Forum, Inc. The Licensor of the FpML Specifications is the International Swaps and Derivatives Association. Copyright © 2006 National Automated Clearinghouse Association. Open Financial Exchange Specification © 2006 by its publishers: CheckFree Corp., Intuit Inc., and Microsoft Corporation. Copyright © SAP AG 2006, Copyright © S.W.I.F.T. SCRL, Avenue Adele, 1, B-1310 La Hulpe, Belgium 2005. The Licensor of the TWIST Standards Specifications is the to be formed TWIST Standards Foundation. All rights reserved by all listed parties.

The FpML Specifications Version 2.0 referenced in the Sterling Commerce Software or related documentation are subject to the FpML Public License; you may not use the FpML Specifications except in compliance with the FpML Public License. You may obtain a copy of the FpML Public License at <http://www.FpML.org>. The FpML Specifications distributed under the FpML Public License are distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the FpML Public License for the specific language governing rights and limitations under the FpML Public License. The Licensor of the FpML Specifications is the International Swaps and Derivatives Association, Inc. All rights reserved.

The Target2 Standards are available from the European Central Bank through its website, <http://www.ecb.int/paym/target/target2/html/index.en.html>. Additionally, the Target2 Standards may be obtained online at <http://www.dnb.nl/dnb/pagina.jsp?pid=tcm:13-46726-64>.

The TWIST Standards Specifications referenced in the Sterling Commerce Software or related documentation are subject to the TWIST Standards Public License; you may not use the TWIST Standards Specifications except in compliance with the TWIST Standards Public License. The TWIST Specifications distributed under the TWIST Public License are distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the TWIST Standards Public License for the specific language governing rights and limitations under the TWIST Standards Public License. The Licensor of the TWIST Standards Specifications is the, to be formed, TWIST Standards Foundation, all rights reserved.

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 <install_dir>/Readme.html.

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

<http://www.sun.com/software/xml/developers/xsdlb2>

<http://www.dhtmlab.com/>

<http://java.sun.com/j2se/downloads.html>

<http://java.sun.com/products/jsse/index-103.html>

<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.

JBoss Software

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

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

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

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

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

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>/rcpdependencies/windows/eclipse/notice.html,
<install_dir>/rcpdependencies/windows/eclipse/plugins/notice.html,
<install_dir>/rcpdependencies/gtk.linux.x86/eclipse/notice.html, and
<install_dir>/rcpdependencies/gtk.linux.x86/eclipse/plugins/notice.html.

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

ICE SOFTWARE and TEE UTILITY SOFTWARE

The Sterling Commerce Software is distributed on the same storage media as the ICE Software (Copyright © 1997 ICE Engineering, Inc./Timothy Gerard Endres.) ("ICE Software") and the Tee Utility Software (Copyright © 2002 Karl M. Syring) ("Tee Utility Software"). The ICE Software and the Tee Utility Software are independent from and not linked or compiled with the Sterling Commerce Software.

The ICE Software and Tee Utility Software are free software products which can be distributed and/or modified under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License or any later version.

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

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

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

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

IEEMU.JS

The Sterling Commerce Software is distributed with or on the same storage media as the IEEMU.js (Copyright (c) 1999-2007 Erik Arvidsson and Emil A. Eklund.) ("IEEMU Software"). Created by Erik Arvidsson and Emil A. Eklund <http://webfx.eae.net/contact.html#erik> for WebFX (<http://webfx.eae.net/>). The IEEMU Software is distributed under the terms of the MOZILLA PUBLIC LICENSE Version 1.1. A copy of this license is located at <install_dir>/3rdParty/ieemu license.doc. The IEEMU.js code is distributed in source form and is located at <install_dir>/repository/eardata/platform/war/yfcscripsts/ieemu.js. 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 IEEMU.js code (and not to the Sterling Commerce Software or any other Third Party Software):

"The contents of the file located at <install_dir>/3rdParty/ieemu license.doc 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 ieemu.js. The Initial Developer is Erik Arvidsson and Emil A. Eklund. Portions created by _____None Listed_____ and the Initial Developer are Copyright © 1999-2007. All Rights Reserved. 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 IEEMU Software and does not apply to the Sterling Commerce Software, or any other Third Party Software.

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.

MYSQL SOFTWARE

The Sterling Commerce Software is distributed on the same storage media as the MySQL Software (Copyright © 1995-2002 MySQL AB) ("MySQL Software"). Before installing the MySQL Software, the terms and conditions of the MySQL license must be accepted.

A copy of the MySQL license is provided at <install_dir>/mysql/MySQL_License.txt. This license only applies to the MySQL Software and does not apply to the Sterling Commerce Software, or any other Third Party Licensor Software.

THE APACHE SOFTWARE FOUNDATION 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, Axis, Apache Commons Lang, Apache Jakarta Commons Collections, Apache Jakarta Commons Pool, Apache Jakarta ORO, Xerces version 2.7, Apache Log4J, Apache SOAP, and Apache Xalan 2.7.0 (collectively, "Apache 2.0 Software"). Apache 2.0 Software is free software which is distributed under the terms of the Apache License Version 2.0. A copy of License Version 2.0 is found in the following directory files for the individual pieces of the Apache 2.0 Software:

```
<install_dir>/ant/Ant_License.txt,  
<install_dir>/jar/antInstaller/0_8/antinstaller_License.txt  
<install_dir>/jar/commons_pool/1_2/Commons_License.txt  
<install_dir>/jar/jakarta_oro/2_0_8/JakartaOro_License.txt  
<install_dir>/jar/log4j/1_2_11/LOG4J_License.txt  
<install_dir>/Xalan_License.txt  
<install_dir>/jar/soap/2_3_1/Apache_SOAP_License.txt  
<install_dir>/jar/commons_collections/2_1/Commons_License.txt  
<install_dir>/jar/commons_lang/2_1/Commons_Lang_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.

Rico Software

The Sterling Commerce Software is also distributed with or on the same storage media as the Rico.js software (Copyright © 2005 Sabre Airline Solutions) ("Rico Software"). Rico Software is free software which is distributed under the terms of the Apache License Version 2.0. A copy of License Version 2.0 is found <install_dir>/3rdParty/rico license.doc.

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

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

SUN MICROSYSTEMS

The Sterling Commerce Software is distributed with or on the same storage media as the following software products (or components thereof): Sun Activation Framework, Sun JSSE, Sun JNet, and Sun JavaMail (collectively, "Sun Software"). Sun Software is free software which is distributed under the terms of the Sun Microsystems, Inc. Binary Code License Agreement ("BCLA"). A copy of the specific BCLA is found in the following directory files for the individual pieces of the Sun Software:

SUN JSSE and JNET JARS - <install_dir>/noapp/lib
SUN JavaMail - <install_dir>/jar/javamail/1_3_2

The licenses relating to the following Sun products are included in the directory files located at:

SUN COMM JAR - <install_dir>/repository/eardata/platform/war/yfscommon
SUN ACTIVATION JAR - <install_dir>/jar/jaf/1_0_2

The Sterling Commerce Software is also distributed with or on the same storage media as the Web-app_2_3.dtd software (Copyright © 2007 Sun Microsystems, Inc.) ("Web-App Software"). Web-App Software is free software which is distributed under the terms of the Common Development and Distribution License ("CDDL"). A copy of the CDDL is found in
<install_dir>/repository/eardata/platform/war/WEB-INF/web_app_licence.txt

The source code for the Web-App Software may be found at:
<install_dir>/repository/eardata/platform/war/WEB-INF/web-app+2_3.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.

W3C Software

The Sterling Commerce Software is distributed on the same storage media as the W3C Software to which the following notice applies:

W3C XML Schema

Copyright © 1994-2007 World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved. This work is distributed under the W3C® Software License [1] in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

[1] <http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>

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

Contents

Preface

Intended Audience	xxi
Structure	xxi
Sterling Multi-Channel Fulfillment Solution Documentation	xxiii
Conventions	xxv

1 Getting Started

1.1 Before You Begin	1
1.2 Installation Checklist	1

2 System Requirements

2.1 Technical Stack Matrix	4
2.2 Database Tier Requirements	4
2.3 Application Server Requirements	5
2.4 Utility Requirements	8
2.5 Internet Browser and Plugin Requirements	9
2.6 Third-Party Software Requirements	9
2.7 Sterling Multi-Channel Fulfillment Solution Mobile Application Requirements. 10	
2.8 Sterling Rich Client Application Requirements	12
2.8.1 Operating System Requirements	12
2.8.2 Minimum Hardware Requirements	13
2.8.3 Sterling Multi-Channel Fulfillment Solution Plugin Requirements	13
2.8.4 Third-Party Software Requirements	14

2.9	Disk Space Requirements for Installation	15
-----	--	----

3 Creating a Security Plan

3.1	Planning Your Deployment Architecture	18
3.1.1	Current Security Infrastructure Analysis	18
3.1.2	Authentication and Authorization	19
3.1.3	Data Encryption	19
3.1.4	Network Topology	20
3.1.4.1	Accessing Over the Public Internet.....	20
3.1.4.2	Deployment Over a Virtual Private Network.....	20
3.1.4.3	Deployment Over a Local Area Network	21
3.2	Java Protocol Security Measures.....	21
3.2.1	Disabling Java Protocols	21
3.2.2	Securing Java Protocols.....	23
3.3	Web Security	25
3.3.1	Post Installation Recommendations.....	25
3.3.2	Session Security.....	26
3.3.3	Operating System Permissions	26
3.3.4	Documentation	27
3.3.5	Routing.....	27
3.3.6	Web Server Executables.....	27
3.4	Database Security	27
3.4.1	Credit Card Encryption	28
3.5	Internet Explorer Security Settings	28
3.5.1	Configuring Browser Security Settings.....	28
3.5.2	Adding the Sterling Multi-Channel Fulfillment Solution as a Trusted Web site	31

4 Installing and Configuring Application Tier Software

4.1	Installing Your Application Server	33
4.2	Installing and Configuring Application Server Utilities	34
4.2.1	Installing JDK Upgrades	34
4.3	Installing and Configuring Your Proxy Server.....	34
4.3.1	Configuring Proxy Server for SSL or HTTPS	35

4.4	Setting Up the Image Server	37
-----	-----------------------------------	----

5 Installing and Configuring Database Tier Software on UNIX/Linux

5.1	Database Sizing	41
5.1.1	Capacity Planning.....	41
5.1.2	Disk Estimation for the Distributed Order Management Module	42
5.1.3	Disk Estimation for the Networked Warehouse Management System Module	45
5.1.4	Tracking and Estimating Future Disk Requirements	51
5.2	Installing Oracle (UNIX/Linux)	51
5.2.1	Oracle Database User Privileges	53
5.2.2	Configuring an Oracle Database for Production	55
5.2.2.1	Enabling the Text Search Feature	57
5.2.3	Using an Oracle Database Server	58
5.2.3.1	Setting Database Parameters in Oracle.....	59
5.2.3.2	Rolling Back or Undoing Changes in Oracle	59
5.2.4	Installing the JDBC Driver in Oracle	59
5.3	Installing DB2 (UNIX/Linux).....	60
5.3.1	DB2 Database User Privileges	60
5.3.2	Configuring a DB2 Database for Production.....	60
5.3.2.1	Enabling the Text Search Feature	62
5.3.3	Using a DB2 Database Server	63
5.3.3.1	Installing Client Components, Compilers, and Fix Pack	64
5.3.3.2	Setting Parameters for DB2	64
5.3.3.3	Installing JDBC Drivers for DB2	64

6 Installing and Configuring Database Tier Software on Windows

6.1	Creating and Configuring the Database Server (Windows)	67
6.2	Database Sizing	68
6.2.1	Capacity Planning.....	68
6.2.2	Disk Estimation for the Distributed Order Management Module	69
6.2.3	Disk Estimation for the Networked Warehouse Management System Module	72

6.2.4	Tracking and Estimating Future Disk Requirements	78
6.3	Installing Microsoft SQL Server 2000/2005 (Windows).....	78
6.3.1	Setting Database Parameters in SQL Server.....	79
6.3.2	SQL Server Database User Privileges	79
6.3.3	Configuring a SQL Server Database for a Production Environment.....	79
6.3.3.1	Running Scripts for a SQL Server Database.....	79
6.3.3.2	Enabling the Text Search Feature	80
6.3.4	Installing the JDBC Driver in SQL Server	81
6.4	Installing Oracle (Windows)	82
6.4.1	Oracle Database User Privileges	83
6.4.2	Configuring an Oracle Database for Production (Windows)	85
6.4.2.1	Enabling the Text Search Feature	86
6.4.3	Using an Oracle Database Server (Windows)	88
6.4.3.1	Setting Database Parameters in Oracle.....	89
6.4.3.2	Rolling Back or Undoing Changes in Oracle	89
6.4.4	Installing the JDBC Driver in Oracle (Windows).....	89

7 Installing in UNIX or Linux Environments

7.1	Before You Begin.....	91
7.2	Preinstallation Setup for a UNIX or Linux Environment.....	91
7.2.1	Key Terms (UNIX/Linux)	92
7.2.2	Checklist for UNIX or Linux Preinstallation.....	92
7.2.3	Checking System Requirements (UNIX/Linux)	96
7.2.4	Installing the Java Software Development Kit (UNIX/Linux).....	96
7.2.5	Creating a UNIX Account	96
7.2.6	Applying Database Definition Language (DDL) Statements (UNIX/Linux) .	96
7.2.7	About Silent Installations (UNIX/Linux)	97
7.2.7.1	Creating the Silent Installation File	97
7.3	Installing Sterling Multi-Channel Fulfillment Solution in a UNIX or Linux Environment.....	98
7.3.1	Before You Install: For AIX Installations Only	99
7.3.2	Running the Installation Program in UNIX or Linux (GUI-Based).....	99
7.3.3	Running the Installation Program in UNIX or Linux (Text-Based).....	104

7.3.4	Running the Installation Program in UNIX or Linux (from Manually Edited Silent Install File)	108
-------	--	-----

8 Installing in a Windows Environment

8.1	Before You Begin	111
8.2	Preinstallation Setup for a Windows Environment	111
8.2.1	Key Terms (Windows)	112
8.2.2	Checklist for Windows Preinstallation	112
8.2.3	Checking System Requirements (Windows)	114
8.2.4	Installing the Java Software Development Kit (Windows)	114
8.2.5	Applying Database Definition Language (DDL) Statements (Windows)	115
8.2.6	About Silent Installations (Windows)	115
8.2.6.1	Creating the Silent Installation File	116
8.3	Installing Sterling Multi-Channel Fulfillment Solution in a Windows Environment.....	118
8.3.1	Running the Installation Program in Windows (GUI-Based)	119
8.3.2	Running the Installation Program in Windows (Manually Edited Silent Install File)	122

9 Installing the Print Server

9.1	Installation of Software Components	125
9.2	Define Printers on Software	126
9.3	Define Printers for the Sterling WMS Installation.....	126
9.4	Copying the Sterling WMS Standard Label Formats	127
9.5	Installation of JasperReports.....	127

10 Installing the Weighing Scale

10.1	Installation of the Weighing Scale.....	129
------	---	-----

11 Installing the Sterling Multi-Channel Fulfillment Solution Mobile Application

11.1	Installation on Mobile Terminals	132
11.1.1	Ensuring Re-Installation on Cold Boot	135

11.1.1.1	On a PocketPC Mobile Terminal	135
11.1.1.2	On a Symbol VRC7900 WinCE Mobile Terminal	136
11.1.1.3	On a Denso BHT400B Win CE 5.0 Mobile Terminal.....	137
11.2	Installing on VT220 Mobile Terminals	138
11.2.1	Installing ncurses	139
11.2.2	Installing libiconv on HP-UX Itanium	139
11.2.3	Installing the Sterling Multi-Channel Fulfillment Solution Mobile Application on VT220 Mobile Terminals.....	139

12 Configuring Properties

12.1	Initial Settings for Properties Files	144
12.2	Overriding Properties	145
12.2.1	Using the Property Parameters File (sandbox.cfg)	146
12.2.2	Using the Customer_Overrides.properties File	146
12.2.2.1	Locating the Property File Name Prefix	147
12.2.2.2	Example 1: Overriding a property from yfs.properties.in	148
12.2.2.3	Example 2: Overriding a property from dbclassCache.properties ..	148
12.3	Properties for LDAP User Authentication	149
12.4	Properties to Prevent Cross-Site Script Vulnerabilities.....	152
12.5	Logging Configuration	152
12.6	Enabling Different Properties for Individual Processes.....	154

13 Configuring Utilities

13.1	Installation Utilities.....	157
13.1.1	Creating Database Schemas and Loading Factory Defaults After Installation.....	158
13.1.1.1	Updating Properties Files After Installation.....	159
13.1.1.2	Verifying the Database Schema.....	160
13.1.1.3	Loading the Sterling Multi-Channel Fulfillment Solution Database Factory Defaults After Installation.....	165
13.1.2	Installing Third-Party JAR Files	166
13.2	Upgrade Utilities	167
13.2.1	Data Migrator	167
13.2.1.1	Data Migrator Task Definition Files.....	167
13.2.1.2	Data Migrator Restart File.....	169

13.2.1.3	Data Migrator Log Files	169
13.3	Development Utilities.....	169
13.4	Runtime Utilities	170
13.4.1	Setting Up the Runtime Utilities.....	171

14 Deploying the Sterling Multi-Channel Fulfillment Solution

14.1	Setting Up the WebLogic Application Server	176
14.1.1	Setting Up the WebLogic Script File	176
14.1.2	Configuring WebLogic for the Sterling Multi-Channel Fulfillment Solution. 178	
14.1.3	Disabling Instrumented Stack Traces in WebLogic.....	178
14.1.4	Setting Up WebLogic to Display Barcodes and Graphs	179
14.1.5	Setting Up WebLogic to Use HTTP In-Memory Session Replication	179
14.2	Building the Enterprise Archive (EAR) Package (WebLogic).....	180
14.2.1	Deploying the Online Help (WebLogic).....	181
14.2.2	Preparing to Build Web Services (WebLogic)	181
14.2.3	Including Custom Classes (WebLogic)	183
14.2.4	Creating the EAR (WebLogic)	183
14.2.4.1	Installing and Deploying Sterling Multi-Channel Fulfillment Solution on Different Servers	184
14.2.5	Precompiling the WAR File (WebLogic)	185
14.3	Deploying the Enterprise Archive (EAR) (WebLogic)	185
14.4	Setting Up the WebSphere Application Server	185
14.4.1	Application Clients Invoking the Sterling Multi-Channel Fulfillment Solution EJBs.....	186
14.4.2	Configuring WebSphere JVM Settings	187
14.4.3	Configuring WebSphere for Running the Sterling Multi-Channel Fulfillment Solution Agents	188
14.4.4	Setting Up WebSphere to Display Barcodes and Graphs.....	189
14.4.5	Configuring DataSource Connection Pooling on WebSphere	190
14.5	Building the Enterprise Archive (EAR) Package (WebSphere)	191
14.5.1	Deploying the Online Help (WebSphere)	191
14.5.2	Preparing to Build Web Services (WebSphere).....	192
14.5.3	Including Custom Classes (WebSphere).....	193
14.5.4	Creating the EAR (WebSphere)	193

14.5.4.1	Installing and Deploying Sterling Multi-Channel Fulfillment Solution on Different Servers	194
14.5.5	Precompiling the WAR File (WebSphere)	194
14.6	Deploying the Enterprise Archive (EAR) (using the WebSphere Admin Console)	194
14.7	Setting Up the JBoss Application Server	196
14.7.1	Setting Up JBoss to Display Barcodes and Graphs	197
14.8	Building the Enterprise Archive (EAR) Package (JBoss)	197
14.8.1	Deploying the Online Help (JBoss)	198
14.8.2	Preparing to Build Web Services (JBoss)	198
14.8.3	Including Custom Classes (JBoss)	200
14.8.4	Creating the EAR (JBoss)	200
14.8.4.1	Installing and Deploying Sterling Multi-Channel Fulfillment Solution on Different Servers	202
14.8.5	Precompiling the WAR File (JBoss)	202
14.9	Deploying the Enterprise Archive (EAR) (JBoss)	202
14.10	Setting the Client Character Display	203
14.11	Clearing Browser and Java Plugin Caches	203
14.12	Statistics Monitoring	204

15 Deploying and Updating the Sterling Rich Client Applications

15.1	Before You Begin	205
15.2	Deploying the Sterling Rich Client Application	207
15.2.1	Creating the RCP_EXTN_FOLDER Folder	207
15.2.1.1	Caching Data Types Locally in the Sterling RCP Based-PCA Client	209
15.2.1.2	Supported Browser Version on Red Hat Linux Workstation 4	210
15.2.1.3	Supported Browser Version on Suse Linux Desktop 10	210
15.2.1.4	Installing Browser Plugins on Linux	210
15.2.1.5	Installing the Flash Player Plugin	211
15.2.1.6	Installing Mozplugger	211
15.2.2	Configuring Locations	211
15.2.2.1	Creating and Configuring a New locations.ycfg XML File	212
15.2.2.2	Modifying the locations.ycfg.sample XML File	213
15.2.3	Localizing Bundle and Theme Files	214
15.2.4	Enabling HTTPS	214

15.2.5	Applying Updates	215
15.2.5.1	Type of Updates	216
15.2.6	Running the Ant Script	217
15.3	Location Configuration Settings	219
15.4	Configuring Connection Settings	221
15.4.1	Configuring Connection Settings for Fetching Images from the Server	223
15.4.2	Configuring Connection Settings for HTTPS Connection	226
15.4.3	Configuring Connection Settings for Making Selective SSL Calls	227
15.4.4	Configuring Connection Settings for Online Help	228
15.5	Security Certificates	228
15.6	Adding Secure APIs for Making Selective SSL Calls	229
15.7	Compression in the Sterling Rich Client Platform	230

A Property Files Reference

A.1	Making Changes to Properties	233
A.2	Setting the Database Connection Properties	234
A.3	yfs.properties*	234
A.4	dbclassCache.properties.*	258
A.5	Sandbox.cfg	259

Index

Preface

This manual explains how to install the various components of the Sterling Multi-Channel Fulfillment Solution and contains information relevant to new installs and upgrades of the Sterling Multi-Channel Fulfillment Solution. It also describes the major tools and components of the Sterling Multi-Channel Fulfillment Solution and provides information about how to set them up in a typical implementation.

Intended Audience

This manual provides installation and administration information for individuals responsible for installing and maintaining the Sterling Multi-Channel Fulfillment Solution.

Structure

This document contains the following sections:

Chapter 1, "Getting Started"

This chapter provides a high-level introduction to the tasks involved in installing the Sterling Multi-Channel Fulfillment Solution.

Chapter 2, "System Requirements"

This chapter describes the hardware and software requirements for installing the Sterling Multi-Channel Fulfillment Solution.

Chapter 3, "Creating a Security Plan"

This chapter provides security recommendations and guidelines.

Chapter 4, "Installing and Configuring Application Tier Software"

This chapter provides information directing you to installation instructions for specific application server software.

Chapter 5, "Installing and Configuring Database Tier Software on UNIX/Linux"

This chapter provides step-by-step instructions for installing and configuring your database server to run the Sterling Multi-Channel Fulfillment Solution on UNIX or Linux operating systems.

Chapter 6, "Installing and Configuring Database Tier Software on Windows"

This chapter provides step-by-step instructions for installing and configuring your database server to run the Sterling Multi-Channel Fulfillment Solution on Windows operating systems.

Chapter 7, "Installing in UNIX or Linux Environments"

This chapter provides step-by-step instructions for installing the Sterling Multi-Channel Fulfillment Solution on UNIX or Linux operating systems.

Chapter 8, "Installing in a Windows Environment"

This chapter provides step-by-step instructions for installing the Sterling Multi-Channel Fulfillment Solution on Windows operating systems.

Chapter 9, "Installing the Print Server"

This chapter describes the Sterling WMS-specific settings for the installation and configuration of the Software Label Manager (LLM) and Software Print Server (LPS).

Chapter 10, "Installing the Weighing Scale"

This chapter describes how to install the weighing scale software used by the Sterling WMS.

Chapter 11, "Installing the Sterling Multi-Channel Fulfillment Solution Mobile Application"

This chapter describes how to install the Sterling Multi-Channel Fulfillment Solution Mobile Application on PocketPC, WinCE, and VT220 mobile devices.

Chapter 12, "Configuring Properties"

This chapter describes how to configure your Sterling Multi-Channel Fulfillment Solution implementation after installation for minimal operations and optional components such as an LDAP server for user authentication.

Chapter 13, "Configuring Utilities"

This chapter describes how to configure the utilities provided with the Sterling Multi-Channel Fulfillment Solution, such as the installation, runtime, and migration script files.

Chapter 14, "Deploying the Sterling Multi-Channel Fulfillment Solution"

This chapter describes how to deploy the Sterling Multi-Channel Fulfillment Solution on an application server.

Chapter 15, "Deploying and Updating the Sterling Rich Client Applications"

This chapter explains how to deploy and update the Sterling Rich Client applications such as Customer Order Management (COM) and Store Operations (SOP) Packaged Composite Applications (PCAs) in different geographical locations.

Sterling Multi-Channel Fulfillment Solution Documentation

For more information about the Sterling Multi-Channel Fulfillment Solution® components, see the following manuals:

- *Sterling Multi-Channel Fulfillment Solution® Release Notes*
- *Sterling Selling and Fulfillment Suite® Release Notes*
- *Sterling Multi-Channel Fulfillment Solution® Installation Guide*
- *Sterling Multi-Channel Fulfillment Solution® Upgrade Guide*
- *Sterling Multi-Channel Fulfillment Solution® Configuration Deployment Tool Guide*
- *Sterling Multi-Channel Fulfillment Solution® Performance Management Guide*

- *Sterling Multi-Channel Fulfillment Solution® High Availability Guide*
- *Sterling Multi-Channel Fulfillment Solution® System Management Guide*
- *Sterling Multi-Channel Fulfillment Solution® Localization Guide*
- *Sterling Multi-Channel Fulfillment Solution® Customization Guide*
- *Sterling Multi-Channel Fulfillment Solution® Integration Guide*
- *Sterling Selling and Fulfillment Suite® Integration Guide*
- *Sterling Multi-Channel Fulfillment Solution® Product Concepts*
- *Sterling Warehouse Management System® Concepts Guide*
- *Sterling Multi-Channel Fulfillment Solution Platform® Configuration Guide*
- *Sterling Distributed Order Management® Configuration Guide*
- *Sterling Supply Collaboration® Configuration Guide*
- *Sterling Global Inventory Visibility® Configuration Guide*
- *Sterling Product Management® Configuration Guide*
- *Sterling Logistics Management® Configuration Guide*
- *Sterling Reverse Logistics® Configuration Guide*
- *Sterling Warehouse Management System® Configuration Guide*
- *Sterling Multi-Channel Fulfillment Solution 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*
- *Sterling Multi-Channel Fulfillment Solution Mobile Application® User Guide*
- *Sterling Multi-Channel Fulfillment Solution Analytics® Guide*

- *Sterling Multi-Channel Fulfillment Solution® Javadocs*
- *Sterling Multi-Channel Fulfillment Solution® Glossary*
- *Sterling Parcel Carrier Adapter® Guide*

Conventions

In this manual, Windows refers to all supported Windows operating systems.

The following conventions may be used in this manual:

Convention	Meaning
. . .	Ellipsis represents information that has been omitted.
< >	Angle brackets indicate user-supplied input.
mono-spaced text	Mono-spaced text indicates a file name, directory path, attribute name, or an inline code example or command.
/ or \	Slashes and backslashes are file separators for Windows, UNIX, and Linux operating systems. The file separator for the Windows operating system is "\" and the file separator for UNIX and Linux systems is "/". The UNIX convention is used unless otherwise mentioned.
<INSTALL_DIR>	User-supplied location of the Sterling Multi-Channel Fulfillment Solution installation directory. This is only applicable for Release 8.0 or above.
<INSTALL_DIR_OLD>	User-supplied location of the Sterling Multi-Channel Fulfillment Solution installation directory for previously installed releases. This is only applicable for Release 8.0 or above.
<YANTRA_HOME>	User-supplied location of the Sterling Supply Chain Applications installation directory. This is only applicable for Release 7.7, 7.9, and 7.11.
<YANTRA_HOME_OLD>	User-supplied location of the Sterling Supply Chain Applications installation directory for previously installed releases. This is only applicable for Releases 7.7, 7.9, and 7.11.

Convention	Meaning
<YFS_HOME>	<p>For releases 7.3, 7.5, and 7.5 SP1, this is the user-supplied location of the Sterling Supply Chain Applications installation directory.</p> <p>For releases 7.7, 7.9, and 7.11, this is the user-supplied location of the <YANTRA_HOME>/Runtime directory.</p> <p>For release 8.0, the <YANTRA_HOME>/Runtime directory is no longer used and this is the same location as <INSTALL_DIR>.</p>
<YFS_HOME_OLD>	This is the <YANTRA_HOME>/Runtime directory of previously installed releases. This is only applicable for Releases 7.7, 7.9, and 7.11.
<ANALYTICS_HOME>	<p>User-supplied location of the Sterling Multi-Channel Fulfillment Solution Analytics installation directory.</p> <p>Note: This convention is used only in the <i>Sterling Multi-Channel Fulfillment Solution Analytics Guide</i>.</p>
<COGNOS_HOME>	<p>User-supplied location of the Cognos installation directory.</p> <p>Note: This convention is used only in the <i>Sterling Multi-Channel Fulfillment Solution Analytics Guide</i>.</p>
<MQ_JAVA_INSTALL_PATH>	<p>User-supplied location of the IBM WebSphere MQ Java components installation directory.</p> <p>Note: This convention is used only in the <i>Sterling Multi-Channel Fulfillment Solution System Management Guide</i>.</p>
<DB>	Refers to the Oracle, DB2, or MSSQL depending on the database server.
<DB_TYPE>	Depending on the database used, considers the value oracle, db2, or sqlserver.

Getting Started

This chapter provides a high-level introduction and checklist for the tasks required to install the Sterling Multi-Channel Fulfillment Solution.

1.1 Before You Begin

Before you begin installing the Sterling Multi-Channel Fulfillment Solution, read this guide thoroughly. Then define your processes for handling the following:

- Development and Test Environments
- Security Strategy
- Change Management Strategy
- Development and Test Procedures
- Rollback Strategy
- Upgrades and Maintenance Strategy

In addition, before starting the installation process, read the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide* which contains information that helps you optimize the performance of your Sterling Multi-Channel Fulfillment Solution.

1.2 Installation Checklist

When installing the components used by the Sterling Multi-Channel Fulfillment Solution, follow the sequence of tasks provided in the ["Checklist"](#) and additional instructions in the chapters of this guide.

During the installation and setup processes, you should also refer to the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*, which is a companion guide and should be used during each step of the process. Doing so can eliminate future problems and help you to troubleshoot errors.

Table 1–1 Installation Checklist

Checklist
<div>1. Ensure that you have the necessary system requirements to install and run the Sterling Multi-Channel Fulfillment Solution.</div> <div>2. Set up your security infrastructure.</div> <div>3. Install and configure your application server.</div> <div>4. Install and configure your WebServer or Proxy Server.</div> <div>5. Install and configure your database software (UNIX) or (Windows).</div> <div>6. Size your database (UNIX) or (Windows).</div> <div>7. Install the Sterling Multi-Channel Fulfillment Solution application (UNIX) or (Windows).</div> <div>8. Optionally install the print server.</div> <div>9. Optionally install the weighing scale software.</div> <div>10. Optionally Install the Sterling Multi-Channel Fulfillment Solution Mobile application.</div> <div>11. Configure the Sterling Multi-Channel Fulfillment Solution properties to use with the database, agent servers, LDAP servers, logging, and so forth.</div> <div>12. Configure the Sterling Multi-Channel Fulfillment Solution utilities for installation, runtime, migration, and production.</div> <div>13. Set up the application server for use with the Sterling Multi-Channel Fulfillment Solution on (WebLogic), (WebSphere), or (JBoss).</div> <div>14. Build your Enterprise Archive (EAR) on (WebLogic), (WebSphere), or (JBoss).</div> <div>15. Deploy the EAR to your application server as appropriate on (WebLogic), (WebSphere), or (JBoss).</div> <div>16. Optionally deploy and update the Sterling Rich Client applications.</div> <div>17. Optionally run the configuration deployment tool to migrate your configuration data. For information about this tool, see the <i>Sterling Multi-Channel Fulfillment Solution Configuration Deployment Tool Guide</i>.</div>

System Requirements

The Sterling Multi-Channel Fulfillment Solution is an n-tier application, using a combination of application server, Web server, and database server software. This chapter lists all the supported operating systems and the required software used in the deployment of the Sterling Multi-Channel Fulfillment Solution. Before installing the Sterling Multi-Channel Fulfillment Solution, verify that you have already installed the applicable software listed in this chapter.

This chapter also provides the information required to complete [Step 1](#) as indicated on the [Installation Checklist](#) on page 1.

Minimum Requirements

This chapter describes the *minimum* supported options for which the Sterling Multi-Channel Fulfillment Solution is already tuned for optimal performance. Your own results are derived from your specific hardware, data volumes, and user activities.

Obtaining Maximum Performance

For information about how to obtain the *maximum* performance from the Sterling Multi-Channel Fulfillment Solution on this supported hardware and software, you must thoroughly read, evaluate, and apply any relevant recommendations described in the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*. For example, the Sterling Multi-Channel Fulfillment Solution predefines a set of indexes, but also expects your Database Administrator to monitor the system and add or remove indexes as necessary.

Note: To avoid data integrity issues, do not remove any *unique* indexes that are provided by the Sterling Multi-Channel Fulfillment Solution.

2.1 Technical Stack Matrix

The Sterling Multi-Channel Fulfillment Solution's technical stack consists of the various tiered hardware and software required by the Sterling Multi-Channel Fulfillment Solution. The technical stack consists of a specific application server, JDK, and database server. You can select various supported configurations from the matrix to create a supported technical stack by following these rules and instructions:

- Select one Database Server.
- Select one Application Server.
- Find a match for the operating system based on your selections of the database and application server.

For example, you may use an Oracle 10g 10.2.0.3 database server on a Sun Solaris 10 operating system with WebSphere on Red Hat Enterprise Linux 5.0 on Intel 32 or 64 bit Xeon or AMD-Opteron processor or SUSE Linux 10 SP1 on 64 bit Xeon or AMD processor.

- Although heterogeneous stacks are supported, when possible, one should consider a homogeneous stack to eliminate having to manage multiple vendors.
- MS SQL Server is supported as a low volume platform with less than 20,000 order lines that can be processed at peak hours.
- Oracle RAC is supported on 2-node or 3-node configurations.

2.2 Database Tier Requirements

This section describes the minimum supported options for the database tier supported in the Sterling Multi-Channel Fulfillment Solution. You can select one database server based on your choice of operating system as shown in [Table 2–1, "Supported Database Tier"](#).

Table 2–1 Supported Database Tier

Database Version	Operating System
Oracle 10g 10.2.0.3	HP-UX 11i v2 on Itanium
	HP-UX 11i v2 on PA-RISC
	IBM AIX 5.3 ML6 <u>only</u> with WebSphere and WebLogic 10
	Sun Solaris 10 on SPARC & AMD Opteron Processors
	Red Hat Enterprise Linux 5.0 /AP 32/64 bit on Xeon and AMD processor
	SUSE Linux Enterprise 10 SP1 on 64 bit Xeon and AMD processor
	Windows 2003 R2
DB2 9.1 FP3	IBM AIX 5.3 ML6 on WebSphere and WebLogic 10 only
MS SQL Server 2000 SP4 or 2005 SP2	Windows 2003 R2 (with Microsoft driver)
Note: Oracle RAC is supported on 2-node or 3-node configurations.	

2.3 Application Server Requirements

This section describes the minimum supported requirements for the agent and application server tier. You can select one application server based on your choice of operating system and Java Messaging Service (JMS) as shown in [Table 2–2](#).

Note: Install the JDK that is shipped with your application server, unless otherwise noted.

Note: 32-bit and 64-bit JDKs are supported for running application servers and Runtime Utilities.

Table 2–2 Supported Application Server Tier

Application Server	Operating System	JMS	JDK
BEA WebLogic 9.2 or 10.x	HP-UX 11i v2 on PA-RISC	Default JMS server that comes with the application server	HP JDK 5.0.11
	HP-UX 11i v2 on Itanium	Default JMS server that comes with the application server	HP JDK 5.0.11
	IBM AIX 5.3 ML6	Default JMS server that comes with the application server	IBM JDK 5.0 (SR4)
	Sun Solaris 10 on SPARC (Sun or Fujitsu) or AMD-Opteron processor	Default JMS server that comes with the application server	Sun JDK 5.0_11 with the Java HotSpot™ Client and Server VMs
	SUSE Linux Enterprise 10 SP1 on 64 bit Xeon or AMD-Opteron processor	Default JMS server that comes with the application server	JRockit 5.0_11
	Red Hat Enterprise Linux 5.0 /AP 32/64 bit on Xeon or AMD processor	Default JMS server that comes with the application server	JRockit 5.0_11
	Windows 2003 R2	Default JMS server that comes with the application server	Sun JDK 5.0_11, JRockit 5.0_11

Table 2–2 Supported Application Server Tier

Application Server	Operating System	JMS	JDK
IBM WebSphere 6.1.0.11	IBM AIX 5.3 ML6	IBM WebSphere MQ 6.0 or default messaging provider that comes with the application server	IBM JDK 5.0 (SR2) (64 bit) shipped with WebSphere
	Sun Solaris 10 on SPARC (Sun or Fujitsu) or AMD-Opteron processor	IBM WebSphere MQ 6.0 or default messaging provider that comes with the application server	IBM JDK 5.0 (SR2) (64 bit) shipped with WebSphere
	Red Hat Enterprise Linux 5.0 /AP 32/64 bit on Xeon or AMD processor	IBM WebSphere MQ 6.0 or default messaging provider that comes with the application server	IBM JDK 5.0 (SR2) (64 bit) shipped with WebSphere
	SUSE Linux Enterprise 10 SP1 on 64 bit Xeon or AMD-Opteron processor	IBM WebSphere MQ 6.0 or default messaging provider that comes with the application server	IBM JDK 5.0 (SR2) (64 bit) shipped with WebSphere
JBoss 4.2.1	Red Hat Enterprise Linux 5.0 /AP 32/64 bit on Xeon or AMD processor	JBoss Messaging or IBM WebSphere MQ 6.0	Sun JDK 5.0_11 with the Java HotSpot™ Client and Server VMs
	Sun Solaris 10 on SPARC (Sun or Fujitsu) or AMD-Opteron processor	JBoss Messaging or IBM WebSphere MQ 6.0	Sun JDK 5.0_11 with the Java HotSpot™ Client and Server VMs

You can install the web server or proxy server that is specified by the application server you choose, to use load balancing and achieve failover.

2.4 Utility Requirements

This section describes which JDKs are required to run utilities on different operating systems.

Table 2–3 JDK Requirements for Utilities

Utility	JDK
Runtime Utilities	
Integration Server	<ul style="list-style-type: none"> For WebSphere, use a JDK from the WebSphere Application Client that has the same version number as the application server. For WebLogic or JBoss, use the same JDK that you use for your application server.
Agent Server	
Agent Trigger	
Installation Utilities	Note: You must specify a 32 bit JDK for use during the Sterling Multi-Channel Fulfillment Solution installation process. 64 bit JDKs are <u>not</u> supported for installation.
Running the Installer	Use the 32 bit version of the JDK that you are using for your agent server.
loadDefaults	Use the 32 bit version of the JDK that you are using for your agent server.
All steps after the Installer through ear precompilation (e.g. merge, ear compilation)	Use the 32 bit version of the JDK that you are using for your agent server.
Upgrade Utilities	
MigrationValidator	Use the 32 bit version of the JDK that you are using for your agent server.
migrator	Use the 32 bit version of the JDK that you are using for your agent server.
Development Utilities	
Configuration Deployment Tool	<ul style="list-style-type: none"> If your application server is WebLogic, use the 32 bit version of Sun JDK 5.0_11 If your application server is WebSphere, use the 32 bit version of Sun JDK 5.0_11 If your application server is JBoss, use the 32 bit version of Sun JDK 5.0_11
Transaction Data Truncation Tool	Use the 32 bit version of the JDK that you are using for your agent server.

2.5 Internet Browser and Plugin Requirements

The minimum requirements for the Internet browser and Java plugin are shown in [Table 2–4](#), "Supported Browsers and the Sterling Multi-Channel Fulfillment Solution Client".

Table 2–4 *Supported Browsers and the Sterling Multi-Channel Fulfillment Solution Client*

Internet Browser	Operating System	JRE
MS Internet Explorer 6.0 SP2	Windows 2000, Windows XP, or Windows 2003 R2	Java Plugin 1.5.0_11
MS Internet Explorer 7.0	Windows XP	Java Plugin 1.5.0_11

Note: By default, the Java plugin memory should be set to 128M. When using the Fulfillment Network Model, set the java plugin memory to 256M. For more information about the Fulfillment Network Model, see the *Sterling Distributed Order Management Configuration Guide*.

Note: For better visibility of the menu options in the Sterling Multi-Channel Fulfillment Solution Consoles, ensure that the dpi setting is 96.

2.6 Third-Party Software Requirements

The requirements for third-party systems such as Cognos Reports, Software Print Server, weighing scale, and so forth are provided in [Table 2–5](#). The software mentioned in the table is supported for all of the operating systems unless otherwise noted.

Table 2–5 Supported Third-Party Software

Name	Version
Build tools	ANT 1.6.5, ANT-CONTRIB (bundled with the Sterling Multi-Channel Fulfillment Solution)
Analytics Reports	Cognos 8 Business Intelligence version 8.2 Cognos ReportNet 1.1 MR3
Analytics Cubes	Cognos 8 Business Intelligence (Windows Only)
ConnectShip	ConnectShip version 5.6
FedEx	FXRS 0764 8.4
FedEx Printer	Eltron LP2844
JasperReports	jasperreports-1.2.0.jar
Loftware Print Server	Loftware 9.0.2.6
Weighing Scale	Mettler-Toledo PS30, PS60, or equivalent.
RFID	Gen 96 Bit Alien Squiggle
FusionCharts	FusionCharts 2.3

2.7 Sterling Multi-Channel Fulfillment Solution Mobile Application Requirements

The minimum system requirements supported for installing the Sterling Multi-Channel Fulfillment Solution Mobile Application are shown in [Table 2–6, "Supported Devices to Run the Sterling Multi-Channel Fulfillment Solution Mobile Application"](#).

Table 2–6 Supported Devices to Run the Sterling Multi-Channel Fulfillment Solution Mobile Application

Terminals	Supported
Mobile Terminals	Pocket PC 2003 OS on .NET CFT 1.0 SP3
	Windows CE 4.1 on .NET 1.0 SP3
	Windows CE 5.0 on .NET 2.0 SP1
	VT220 emulation software
ncurses (VT220 emulation software)	Version 5.3
ncurses build utilities	gcc-3.3.2, bison-1.875, make-3.80, and flex-2.5.4a. These utilities are available in binary format at http://hpux.cs.utah.edu .
Note: The Sterling Multi-Channel Fulfillment Solution is specifically tested with the Symbol PPT8846, Symbol PDT8146, Symbol PPT 8800, and Symbol VRC7900 series; the Denso BHT-400B, Denso BHT-260Q, and Denso BHT-420BW series; and the LXE_VXC001 and LXE_MX7001 series.	

The device requirements such as memory, screen resolution, keys, and network connectivity are described in [Table 2–7, "Device Requirements"](#).

Table 2–7 Device Requirements

Options	Description
Keys	Space, Backspace, Tab, Enter, arrow keys, 0-9, A-Z, a-z, function keys (F1-F12), and special symbols such as !@#\$%^&*()-_+=[]\<>?/.,. The special symbols are required only if the data (for fields such as Item ID, Location ID, and Shipment Number) contains special characters.
Barcode Scanner	This device should be equipped with an integrated barcode scanner and should have the ability to send a TAB character after the scanned data.
Note: Microsoft ActiveSync 3.7 or above is required to synchronize the PC with the mobile terminal to run the Sterling Multi-Channel Fulfillment Solution Mobile Application. Microsoft ActiveSync 3.7 can be installed on any PC. For more information about Microsoft ActiveSync, and related system requirements, go to http://www.microsoft.com .	

Table 2–7 Device Requirements

Options	Description
Screen Resolution	Width: 240, Height: 320. Screens are designed for this resolution. Screen performance on devices with other resolutions may be sub-optimal. For VT220 screens, the Sterling Multi-Channel Fulfillment Solution assumes 8 rows by 20 columns.
Network Connectivity	802.11x (802.11b or higher).
Memory	Minimum of 32MB RAM.
Note: Microsoft ActiveSync 3.7 or above is required to synchronize the PC with the mobile terminal to run the Sterling Multi-Channel Fulfillment Solution Mobile Application. Microsoft ActiveSync 3.7 can be installed on any PC. For more information about Microsoft ActiveSync, and related system requirements, go to http://www.microsoft.com .	

2.8 Sterling Rich Client Application Requirements

This section lists all the supported operating systems and the required software used in the deployment of the Sterling Rich Client application. Before installing the Sterling Rich Client application, verify that you have already installed the applicable software listed in this chapter.

2.8.1 Operating System Requirements

The minimum operating system requirements supported for installing the Sterling Rich Client application are shown in [Table 2–8, "Supported Operating Systems"](#).

Table 2–8 Supported Operating Systems

Name	Configuration
Windows	Windows 2000, Windows XP
Linux	Red Hat Enterprise Linux WS 5 SUSE Linux Enterprise Desktop 10 SP1

Important: For optimal resolution quality of the menu options and other user interface components of the Sterling Rich Client applications, set your system resolution to 96 dpi.

2.8.2 Minimum Hardware Requirements

The minimum hardware requirements for installing the Sterling Rich Client applications are listed in [Table 2–9, "Minimum and Recommended Hardware Requirements"](#).

Table 2–9 Minimum and Recommended Hardware Requirements

Component	Minimum	Recommended
Processor	350 MHz	1 GHz
Memory	512 MB	1 GB

2.8.3 Sterling Multi-Channel Fulfillment Solution Plugin Requirements

The Sterling Multi-Channel Fulfillment Solution plugins supported for installing the Sterling Rich Client application are shown in [Table 2–10, "Supported Sterling Multi-Channel Fulfillment Solution Plugins"](#).

Table 2–10 Supported Sterling Multi-Channel Fulfillment Solution Plugins

Name	Version
Sterling RCP plugin	Sterling RCP Plugin 1.0.0
Sterling RCP Tools plugin	Sterling RCP Tools Plugin 1.1.0

To verify that you have a supported version of the plugins listed in [Table 2–10](#), please follow the steps listed below:

1. Open Eclipse
2. Navigate to Help > About Eclipse SDK
3. Click Plug-in Details
4. Verify that the Plug-ins listed in table [Table 2–11](#) match those that are listed in Eclipse.

Table 2–11 Plug-ins Installed in Eclipse

Name	Plug-in Name
Sterling RCP plugin	Sterling Commerce Rich Client Platform Foundation Plugin 1.0.0
	Sterling Commerce Rich Client Platform Libs Plugin
	Sterling Commerce Rich Client Platform Plugin 1.0.0
Sterling RCP Tools plugin	Sterling Commerce RCP Extensibility Tools 1.1.0
	Sterling Commerce RCP Tools Core Plug-in 1.1.0
	Sterling Commerce RCP Tools UI Editor Plug-in 1.1.0

2.8.4 Third-Party Software Requirements

The third-party software such as Eclipse SDK and its related plugins, JDK, and so forth, are provided in [Table 2–12, "Supported Third-Party Software"](#). The software mentioned in the table are supported for all the operating systems except where noted.

Table 2–12 Supported Third-Party Software

Name	Version								
Eclipse SDK	<div>Eclipse SDK 3.2</div> <div>The following table lists the Eclipse-related plugins and their versions that Sterling Multi-Channel Fulfillment Solution supports:</div> <table><tr><th>Name</th><th>Version</th></tr><tr><td>GEF plugin</td><td>GEF Plugin 3.2, SDK</td></tr><tr><td>EMF plugin</td><td>EMF Plugin 2.2.0, SDK (includes EMF, SDO, XSD)</td></tr><tr><td>VE plugin</td><td>Visual Editor Plugin 1.2, SDK</td></tr></table>	Name	Version	GEF plugin	GEF Plugin 3.2, SDK	EMF plugin	EMF Plugin 2.2.0, SDK (includes EMF, SDO, XSD)	VE plugin	Visual Editor Plugin 1.2, SDK
Name	Version								
GEF plugin	GEF Plugin 3.2, SDK								
EMF plugin	EMF Plugin 2.2.0, SDK (includes EMF, SDO, XSD)								
VE plugin	Visual Editor Plugin 1.2, SDK								
JDK	<div>JDK 1.5.0_11.</div> <div>The JDK is used for both building and extending the client application.</div>								
JRE	<div>JRE 1.5.0_11.</div> <div>The JRE is used to run the Rich Clients.</div>								

To install the Eclipse SDK and its related plugins, go to the following link: <http://www.eclipse.org/downloads/> and download the appropriate version of Eclipse SDK and its related plugins for the appropriate operating system.

To install the JDK/JRE, go to the following link: <http://java.sun.com/downloads/> and download the appropriate version of the JDK.

2.9 Disk Space Requirements for Installation

During the Sterling Multi-Channel Fulfillment Solution installation on Windows, the installer checks for a minimum of 5GB of free disk space on the specified drive. If the system does not meet this requirement, the installation fails.

Creating a Security Plan

This chapter provides security recommendations and guidelines for running the Sterling Multi-Channel Fulfillment Solution. It is intended to help you create a reasonably secure implementation of the application.

This chapter also provides the information required to complete [Step 2](#) indicated on the [Installation Checklist](#) on page 1.

Because we recognize that you may have unique business or operational requirements, the Sterling Multi-Channel Fulfillment Solution does not provide a specific set of instructions that you can follow to completion for creating a security plan. Typically, it is not possible to configure a system solely for security at the detriment of other engineering or business realities.

Given the complicated nature of security, it is recommended that you refer to the following documents:

- <http://www.nsa.gov/snac> for tips on how to harden your operating system, database, and network
- <http://e-docs.bea.com/wls/docs100/secmanage/> for tips on how to secure BEA WebLogic 10
- <http://technet.microsoft.com/en-us/sqlserver/bb331806.aspx> for tips on how to secure Microsoft SQL Server 2005
- The Rhino9 Team, *The Modern Hackers Desk Reference*; available from <http://www.f4.ca/text/mhdr.html>.
- Tom Bialaski and Michael Haines, *Solaris and LDAP Naming Services, Deploying LDAP in the Enterprise*; Prentice Hall PTR, 2001.

3.1 Planning Your Deployment Architecture

Prior to procuring and implementing the hardware and software that make up the Sterling Multi-Channel Fulfillment Solution, you need to plan your deployment architecture by completing the following tasks:

- Conduct an analysis of the current security infrastructure in your organization. For more information on identifying the correct security infrastructure in the Sterling Multi-Channel Fulfillment Solution, see [Section 3.1.1, "Current Security Infrastructure Analysis"](#) on page 18.
- Conduct an analysis of authentication and authorization mechanisms in your organization to identify the steps needed to incorporate them into the Sterling Multi-Channel Fulfillment Solution. For more information on the mechanism used for authentication in the Sterling Multi-Channel Fulfillment Solution, see [Section 3.1.2, "Authentication and Authorization"](#) on page 19.
- Conduct an analysis of your data encryption mechanisms for deploying the Sterling Multi-Channel Fulfillment Solution over the internet. For more information on the different variations of the data encryption mechanisms, refer to [Section 3.1.3, "Data Encryption"](#) on page 19.
- Conduct an analysis of your organization's network topology required to deploy the Sterling Multi-Channel Fulfillment Solution. For more information on the various methods to deploy the application, refer to [Section 3.1.4, "Network Topology"](#) on page 20.

Completing these tasks enables you to:

- Estimate your server requirements.
- List the major security software and hardware needed to implement the Sterling Multi-Channel Fulfillment Solution.

3.1.1 Current Security Infrastructure Analysis

In order to ensure that your Sterling Multi-Channel Fulfillment Solution is a secure Web application, there are many factors involved. Be sure to answer the following questions before you start your Sterling Multi-Channel Fulfillment Solution implementation.

- Does your organization have security personnel? If not, you may wish to seek input from an Internet security company in your area.

- Do you own a network scanner such as Internet Security Systems System Scanner or Internet Scanner? Products like these help you identify common problems with servers that are exposed to the Internet.
- Do you own an intrusion detection system such as Symantec Intruder Alert? This type of product works with your firewall to stop an intrusion before mission-critical data or systems are tampered with.

3.1.2 Authentication and Authorization

Authentication and authorization are vital to security. Due to the constantly changing authentication methodologies, including biometrics, public key infrastructure (PKI), and ever-increasing encryption algorithms, the Sterling Multi-Channel Fulfillment Solution provides documentation on implementing a lightweight directory access protocol (LDAP) or any Java Authentication and Authorization Service (JAAS) compliant security module for authentication. With LDAP user and password management can be centralized. For information on deploying the Sterling Multi-Channel Fulfillment Solution and integrating with LDAP, see [Section 12.3, "Properties for LDAP User Authentication"](#) on page 149. The default authentication mechanism is implemented against the Sterling Multi-Channel Fulfillment Solution database.

3.1.3 Data Encryption

Due to the differences in the nature of businesses, you may implement different security measures when implementing a web application. How you plan to deploy the application and what security measures are taken are unique to each business. Most security measures come at a cost of performance. The Internet is a public network. Sensitive data should be encrypted while traveling across it. Encrypting information that travels across the Internet has an associated cost. If the Sterling Multi-Channel Fulfillment Solution is not to be deployed on the Internet, encryption may not be necessary and the cost is thereby negated.

The data encryption mechanisms recommended for the Sterling Multi-Channel Fulfillment Solution are:

- SSL - 128-bit encryption is the recommended encryption level.
- VPN - 3DES or AES is the recommended encryption algorithm.

3.1.4 Network Topology

Where is the Sterling Multi-Channel Fulfillment Solution being accessed from?

- Public Internet?
- Virtual private network (VPN)?
- Internal Local area network (LAN)?

The Sterling Multi-Channel Fulfillment Solution application is typically implemented as an internal application that is accessible from an Internal Network or across from VPN.

Regardless of which network, we strongly recommend that you use SSL to encrypt all the Sterling Multi-Channel Fulfillment Solution screen requests. SSL processing can be expensive and can add an additional 30% or more processing overhead to each application server transaction. Depending on your transaction volumes, you may want to off-load your SSL processing to specialized devices such as an F5 load-balancer with built-in hardware SSL engines.

3.1.4.1 Accessing Over the Public Internet

If you are accessing the Sterling Multi-Channel Fulfillment Solution over the Public Internet you have to also consider additional security concerns such as denial of service attacks.

3.1.4.2 Deployment Over a Virtual Private Network

If you are deploying the Sterling Multi-Channel Fulfillment Solution over a virtual private network (VPN), the major factor in security and performance is the VPN encryption. Many firewall providers offer encryption and decryption accelerators that can be added directly to their firewalls. Checkpoint's FireWall-1, VPN-1 Accelerator Card II, is an example of this. However, one consideration for purchasing accelerator cards is how many VPN tunnels are needed. You also need to determine if the VPN is being set up for site-to-site implementation or if each individual user opens their own tunnel. If you decide on a site-to-site VPN, typically memory in the firewall is the greatest concern. If each user opens their own tunnel, processor speed is the largest concern.

In many cases the deciding factor is the speed at which your VPN is connected. If you have a T1 line, a single processor machine may suit

your needs. If you plan to deploy over a T3 line, you may wish to consider a multiple-processor machine. Most firewall and VPN vendors can help you size the machine you purchase from them for optimal security and performance.

3.1.4.3 Deployment Over a Local Area Network

If you are deploying the Sterling Multi-Channel Fulfillment Solution over a local area network (LAN), performance should not be an issue. We strongly recommend you SSL all the Sterling Multi-Channel Fulfillment Solution screens even on an Internal Network.

3.2 Java Protocol Security Measures

As with the usage of any protocol technology there are certain associated risks. Sterling Multi-Channel Fulfillment Solution APIs are exposed over various protocols. Therefore, Sterling Commerce strongly recommends that you disable protocols that you do not use.

3.2.1 Disabling Java Protocols

Each of the following sections provide instructions to disable the respective protocols not used.

EJB

To disable Enterprise JavaBeans (EJB) from the Sterling Multi-Channel Fulfillment Solution, comment out the "session" element in the XML descriptor file,

```
<INSTALL_DIR>/repository/eardata/platform/descriptors/<App_Server>/EJB/META-INF/ejb-jar.xml.
```

Important: To avoid an error when deploying the ejb-jar.xml for WebLogic, you must comment out the following session bean of the xml file:

```
<session>
  <display-name> The Sterling Multi-Channel Fulfillment Solution DOM API Session bean
</display-name>
  <ejb-name> interop.services.ejb.InteropEJBApi </ejb-name>
  <home> com.yantra.interop.services.ejb.InteropEJBHome </home>
  <remote> com.yantra.interop.services.ejb.InteropEJBApi </remote>
  <ejb-class> com.yantra.interop.services.ejb.InteropEJBImpl </ejb-class>
  <session-type> Stateless </session-type>
  <transaction-type> Bean </transaction-type>
</session>
```

This session bean is deprecated as of Release 7.7.

HTTP

To disable Hypertext Transfer Protocol (HTTP) as the means to enter API information in the Sterling Multi-Channel Fulfillment Solution, the deployment descriptor needs to be modified. The deployment descriptor, `web.xml`, is defined by the servlet specification from Sun Microsystems. This deployment descriptor can be used to deploy a web application on any J2EE-compliant application server. The deployment descriptors for the Sterling Multi-Channel Fulfillment Solution are stored in the `<INSTALL_DIR>/repository/eardata/yantra/descriptors/<App_Server>/WAR/WEB-INF` directory. The deployment descriptor for the `InteropHttpServlet` needs to be removed from the `web.xml` file to disable the servlet. Remember to remove both the `servlet-name` and the `servlet-mapping` entries from this file.

JMS

In order to use the Java Messaging Service (JMS) features of the Sterling Multi-Channel Fulfillment Solution, there must be a JMS server. There must be queues set up both on the JMS Server and within the Sterling Multi-Channel Fulfillment Solution.

To ensure that JMS is not used without authorization there should be appropriate permissions on the JMS server and in the Sterling Multi-Channel Fulfillment Solution. You can limit the ability of users to enable JMS by disabling permissions using Process Modeling in the Sterling Multi-Channel Fulfillment Solution Configurator. For more

information about enabling and disabling permissions, see the *Sterling Multi-Channel Fulfillment Solution Platform Configuration Guide*.

3.2.2 Securing Java Protocols

Protocols are specified in the `yifclient.properties` file as `LOCAL`. To use a different protocol, add a `yif.yif.apifactory.protocol=<protocol_type>` entry in the `<INSTALL_DIR>/properties/customer_overrides.properties` file. Other valid values for `<protocol_type>` are `HTTP`, `HTTPS`, and `EJB`.

Note: If you use an EJB protocol, you must also add the following property entries to the `<INSTALL_DIR>/properties/customer_overrides.properties` file based on your application server:

For WebLogic:

```
yif.java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory
yif.java.naming.provider.url=t3://<ipaddress>:<port>
```

For WebSphere:

```
yif.java.naming.factory.initial=com.ibm.websphere.naming.WsnInitialContextFactory
yif.java.naming.provider.url=iiop://<ipAddress>:<port>
```

EJB

When the Sterling Multi-Channel Fulfillment Solution APIs are deployed through EJB, they use a Java Naming and Directory Interface (JNDI) lookup for a context to call the EJB Objects. JNDI looks up a context that is a handle to the EJB Object or API. The APIs do not have authentication or authorization. However, security principal and credentials can be supplied by specifying them in the `yifclient.properties` configuration file. The server can be set up to validate the passed security credentials.

The Sterling Multi-Channel Fulfillment Solution HTTP/HTTPS Interface uses JavaServer Pages (JSPs) installed on the application server and

does not need access to JNDI. There are two ways to protect the Sterling Multi-Channel Fulfillment Solution APIs over EJB:

- WebLogic allows JNDI and remote method invocation (RMI) to be tunneled over HTTP. In your architecture there should be a proxy to inspect all the requests for "Sterling Multi-Channel Fulfillment Solution". This ensures that all the requests are for HTML, and not tunneled RMI or JNDI over HTTP.
- If the Sterling Multi-Channel Fulfillment Solution is deployed on WebLogic, a security realm should be set up to protect JNDI resources. This does not affect any screens that are packaged with the Sterling Multi-Channel Fulfillment Solution or any screens that extend the Sterling Multi-Channel Fulfillment Solution.

If the application is deployed on WebSphere or JBoss, you must set up permissions for EJB method. This does not affect any standard screens that are packaged with the Sterling Multi-Channel Fulfillment Solution or the custom screens you create.

Important: If you attempt to run the Sterling Multi-Channel Fulfillment Solution using HTTPS, the Sterling Multi-Channel Fulfillment Solution Configurator does not open.

If a custom user interface is being built using the Sterling Multi-Channel Fulfillment Solution APIs through EJB and not by extending the Sterling Multi-Channel Fulfillment Solution Presentation Framework, you cannot use the client wrapper supplied with the Sterling Multi-Channel Fulfillment Solution because it currently is incapable of passing credentials. This also applies to any use of the `YIFAPIFactory` class.

HTTP API Tester

The HTTP API tester is provided *only* to test APIs in development mode. Authentication and authorization are not used in this utility. If you plan to provide access to this page in production, you should secure access to it.

To secure access to the Sterling Multi-Channel Fulfillment Solution `httpapitester`, the deployment descriptor needs to be modified. The deployment descriptor's `web.xml` is defined by the servlet specification from Sun Microsystems. This deployment descriptor can be used to

deploy a web application on any J2EE-compliant application server. The deployment descriptor for the Sterling Multi-Channel Fulfillment Solution are stored in the

`<INSTALL_DIR>/repository/eardata/yantra/descriptors/<App_Server>/WAR/WEB-INF` directory. By using the security-constraint element with the web-resource-collection element, you can set up authorization to protect this page from unauthorized access. For more information about the `web.xml` deployment descriptor, see the documentation for your application server.

Alternatively, you can simply remove the `yfshttpapi` directory under `<INSTALL_DIR>/webpages` and secure the `/interop/InteropHttpServlet` servlet using the security features provided by your application server.

COM+

The extended Component Object Model (COM+) specification covers security in great detail. Any COM+ object deployed on a server complies with this standard. For information on setting up security for COM+ objects, see *The Microsoft Developers Network* article available at: <http://msdn.microsoft.com/en-us/library/ms681314.aspx>

3.3 Web Security

Sterling Commerce highly recommends that a security audit is made prior to deployment.

Sterling Commerce also recommends that you write log files to several servers. There are several applications that do this with no specific need for the Sterling Multi-Channel Fulfillment Solution to duplicate their efforts. Additionally, products like Symantec's Intruder Alert monitor log files for authentication failures and alert an administrator if a threshold is exceeded.

3.3.1 Post Installation Recommendations

After the installation of the Sterling Multi-Channel Fulfillment Solution, be sure to complete the following for ensured security:

1. Change the password of the default user (admin).

2. Secure the `<INSTALL_DIR>/database`, `<INSTALL_DIR>/repository`, and `<INSTALL_DIR>/installed_data/sscap/components/complete_installation/entity` directories because they expose components of the data model.
3. Sterling Multi-Channel Fulfillment Solution supplies Web pages that help you test your implementation while running the Sterling Multi-Channel Fulfillment Solution in development mode. Remove the `<INSTALL_DIR>/installed_data/platform_afc/files/repository/eardata/platform/war/yfshttpapi/yantrahttpapitester.jsp` and `<INSTALL_DIR>/repository/eardata/platform/war/yfshttpapi/yantrahttpapitester.jsp` files from production systems. If you plan to provide access to this jsps in production, secure access to them as described in ["HTTP API Tester"](#) on page 24.
4. Change permissions on `<INSTALL_DIR>/bin/migrator.*` files to non-executable.

3.3.2 Session Security

Session security is handled by the application server, and is stored in a non-persistent cookie on the client. You should ensure that all transactions with the application server are protected with SSL to prevent session hijacking attacks.

3.3.3 Operating System Permissions

The following files contain confidential information, such as user name and password combinations stored in clear text. These files should be secured through operating system permissions:

- `sandbox.cfg`
- `<appserver>.log`
- `jdbc.properties.in`
- `yfs.properties.in`
- `yifclient.properties.in`

3.3.4 Documentation

All the documentation files for the Sterling Multi-Channel Fulfillment Solution and third-party software should be removed from any production servers.

3.3.5 Routing

Routing should not be enabled on a production web server.

3.3.6 Web Server Executables

Web servers should *not* be run as root. This ensures that if someone compromises any software associated with the deployment through a bug, they don't have root privileges to damage the server. Web servers allow you to access files on their host machines and as root any of those files can be modified for a deeper attack or deleted to make your web servers unavailable.

It is acceptable, although not recommended, to start the web server as root. A proxy server can be used to accept HTTP traffic and redirect it to a port above 1024 on a UNIX system. If a proxy is not available and the web server must be started on port 80 it is necessary to start the web server as root. The web server then calls `setuid` to transfer root privileges to a generic unprivileged account. The web server's configuration file should allow you to specify what user it runs as. Any user may own the binary. The `setuid` bit should not be set on the web server binary.

3.4 Database Security

Set up separate accounts on the database server for installing the Sterling Multi-Channel Fulfillment Solution schema and for accessing the application database.

If using an Oracle database on the production database server, the Oracle parameter `DBLINK_ENCRYPT_LOGIN` in your `init.ora` file should be set to `TRUE`. This ensures that all connections to the database are not sent as clear text.

3.4.1 Credit Card Encryption

If you want to ensure that credit card numbers are encrypted at the database level, you configure that functionality when setting Hub attributes in the Sterling Multi-Channel Fulfillment Solution Configurator. When setting Hub attributes, make sure that the credit card number encrypting option is checked. For more information and specific instructions for setting up security, see the *Sterling Multi-Channel Fulfillment Solution Platform Configuration Guide*.

Sterling Multi-Channel Fulfillment Solution also supplies APIs and user exits to encrypt credit card and other secure information. For more information about these APIs, user exits, and other data encryption, see the *Sterling Multi-Channel Fulfillment Solution Customization Guide* and the *Sterling Multi-Channel Fulfillment Solution Javadocs*.

3.5 Internet Explorer Security Settings

When using the Sterling Multi-Channel Fulfillment Solution without any customizations, you may need to set security or privacy settings for your Internet Explorer in order to obtain the best browser performance.

3.5.1 Configuring Browser Security Settings

To configure the Internet Explorer security and privacy settings:

1. From the Internet Explorer menu, select Tools > Internet Options > Security.
2. Select the Web content zone from which the Sterling Multi-Channel Fulfillment Solution is accessed.
3. Choose Default Level and set the security level to High.
4. Depending on the version of Internet Explorer you have installed, choose Custom Level and set your security settings according to one of the following:
 - [Table 3–1, "Internet Explorer Version 6.0 Security Settings for the Sterling Multi-Channel Fulfillment Solution"](#)
 - [Table 3–2, "Internet Explorer Version 6.0 Privacy Settings for the Sterling Multi-Channel Fulfillment Solution"](#) on page 30

Table 3–1 Internet Explorer Version 6.0 Security Settings for the Sterling Multi-Channel Fulfillment Solution

Internet Explorer Security Setting	Sterling Multi-Channel Fulfillment Solution
ActiveX Controls and Plugins	
Download signed ActiveX controls	Prompt/Enable
Download unsigned ActiveX controls	Disable
Initialize and script ActiveX controls not marked as safe	Disable
Run ActiveX controls and plugins	Prompt/Enable
Script ActiveX controls marked as safe for scripting	Enable
Cookies	
Allow cookies that are stored on your computer	Disable (Enable only if you are using Sterling Multi-Channel Fulfillment Solution Analytics)
Allow per-session cookies	Enable
Downloads	
File download	Enable
Font download	Prompt
Microsoft VM	
Java permissions	High Safety
Miscellaneous	
Access data sources across domains	Disable
Allow META REFRESH	Disable
Display mixed content	Prompt
Do not prompt for client certificate selection when no certificates or only one certificate exists	Disable

Table 3–1 Internet Explorer Version 6.0 Security Settings for the Sterling Multi-Channel Fulfillment Solution

Internet Explorer Security Setting	Sterling Multi-Channel Fulfillment Solution
Drag and drop or copy and paste files	Prompt
Installation of desktop items	Disable
Launching programs and files in an IFRAME	Disable
Navigate sub-frames across different domains	Disable
Software channel permissions	High Safety
Submit non-encrypted form data	Prompt
Userdata persistence	Disable
Scripting	
Active scripting	Enable
Allow paste operations via script	Enable
Scripting of Java applets	Enable
User Authentication	
Logon	Prompt for user name and password

Table 3–2 Internet Explorer Version 6.0 Privacy Settings for the Sterling Multi-Channel Fulfillment Solution

Advanced Privacy Setting	Sterling Multi-Channel Fulfillment Solution
Override automatic cookie handling	Yes
First-Party Cookies	Block
Third-Party Cookies	Block
Always allow session cookies	Yes

3.5.2 Adding the Sterling Multi-Channel Fulfillment Solution as a Trusted Web site

You should set the Sterling Multi-Channel Fulfillment Solution to be recognized as a trusted Web site. Not doing so could cause certain pop-up windows such as date and time selection to display a status bar, thereby hiding certain action buttons.

To add the Sterling Multi-Channel Fulfillment Solution to the list of trusted Web sites:

1. In the Internet Explorer menu bar, select Tools > Internet Options. The Internet Options pop-up window is displayed.
2. In the Internet Options pop-up window, select the Security tab.
3. Click the Trusted Sites icon.
4. Click the Sites action button. The Trusted Sites pop-up window is displayed.
5. In the 'Add this Web site to the zone' text box, enter the server address where the Sterling Multi-Channel Fulfillment Solution Consoles are installed. The port number does not need to be specified.
6. Uncheck the 'Require server verification (https:) for all sites in this zone' checkbox.
7. Click OK. This takes you back to the Internet Options pop-up window.
8. Click OK.

Installing and Configuring Application Tier Software

Before installing an application server, ensure that you have installed the required software mentioned in [Chapter 2, "System Requirements"](#), noting any recommendations supplied by the software provider and by Sterling Commerce, Inc. This chapter supplies information to help you install software on the application server and web server tier.

Note: Before proceeding with the steps in this chapter ensure that you know the precise installation location for the Sterling Multi-Channel Fulfillment Solution (referred to as <INSTALL_DIR>).

This chapter also provides the information required to complete [Step 3](#) and [Step 4](#) indicated on the [Installation Checklist](#) on page 1.

4.1 Installing Your Application Server

Before installing your application server, check the requirements in [Chapter 2, "System Requirements"](#) to make sure you have the applicable hardware and software versions installed.

Install your application server according to the instructions on the product CD-ROM disk:

- If you purchased BEA WebLogic directly from BEA, see the *Installing BEA WebLogic Platform* on the product CD-ROM disk for installation instructions.

- If you purchased WebSphere directly from IBM, see the *WebSphere Installation Guide* on the product CD-ROM disk for installation instructions.
- If you purchased JBoss directly from Red Hat, see the *JBoss Installation Guide* on the product CD-ROM disk for installation instructions.

4.2 Installing and Configuring Application Server Utilities

You need to configure certain application server utilities before installing the Sterling Multi-Channel Fulfillment Solution.

Install the Java Development Kit (JDK) that is shipped with your application server unless otherwise stated in [Chapter 2, "System Requirements"](#). When upgrading the JDK, be sure to set the correct `JAVA_HOME` environment variable and update the `PATH`.

4.2.1 Installing JDK Upgrades

You should install the Java Development Kit (JDK) that is shipped with your application server (unless otherwise specified in [Chapter 2, "System Requirements"](#)). When upgrading the JDK, be sure to set the correct `JAVA_HOME` environment variable and update the `PATH`.

4.3 Installing and Configuring Your Proxy Server

Installing a proxy web server on a dedicated hardware provides:

- Additional network security layers.
- Additional processing power for data encryption protocols.
- Additional options for high availability for your application.

You can install a proxy or Web server to avoid any bottlenecks that may occur when systems try to access the Sterling Multi-Channel Fulfillment Solution installed on your application server. Sterling Commerce recommends that you install and configure the Web server version as specified by your application server provider.

4.3.1 Configuring Proxy Server for SSL or HTTPS

This section explains how to set up a web server as an SSL proxy and a load balancer. The SSL proxy allows the web server to manage the SSL encryption load and pass clear text back to application servers. It also divides the workload among the available application servers using the "round-robin load balancing" algorithm. The web server allows users to use one secure URL to access any number of application servers that run the Sterling applications.

By configuring a proxy server for SSL or HTTPS, an SSL or HTTPS connection is set up between the client and web server, which allows a clear text connection between the web server and application server. This reduces the network traffic between the web server and application server.

For more information about configuring the proxy server for SSL or HTTPS using the Apache HTTP Server and WebSphere, see the IBM documentation.

Using any standard Web browser, the information about configuring the Apache HTTP server as a proxy server and a load balancer using SSL or HTTPS on JBoss can be found at:

http://wiki.jboss.org/wiki/Wiki.jsp?page=UsingMod_proxyWithJBoss.

To configure a proxy server for SSL or HTTPS using the Apache HTTP Server and BEA WebLogic:

1. Install and run the Sterling Multi-Channel Fulfillment Solution on the application servers.
2. Copy the appropriate plug-in to the `/etc/httpd/modules` directory.

For WebLogic 9.2, this is:

```
<WL_HOME>/weblogic92/server/native/<OS>/<processor  
type>/plugin
```

For WebLogic 10.0, this is:

```
<WL_HOME>/wlserver_100/server/plugins/<OS>/<processor  
type>/plugin
```

where `<WL_HOME>` refers to the WebLogic installation directory.

- For i686, copy the WLS plug-in.

- For x86_64, copy the 64-bit plug-in. The 64-bit plug-in must be requested from BEA Customer Service.
 - Ensure the plugin is executable.
3. To enable the WebLogic plug-in for load-balancing using HTTP or HTTPS, modify the `httpd.conf` file and add the following:

```
LoadModule weblogic_module /etc/httpd/modules/<appserver_plugin_file>
```

Note: To enable an SSL, ensure to add "include conf.d/ssl.conf" as instructed by Apache. By default, RHAS3 has "Include conf.d/*", which includes ssl.conf.

For an HTTP proxy, outside any VirtualHost, add the following section:

```
<IfModule mod_weblogic.c>
    WebLogicCluster
    <managed_server1_hostname/IP_address>,<managed_server2_hostname/IP_address>
    DynamicServerList OFF
    Debug ON
    IdemPotent OFF
</IfModule>
<Location /yantra>
    SetHandler weblogic-handler
</Location>
```

4. Modify the `ssl.conf` file and add the following lines to the `<VirtualHost _default_:443>` section:

```
<IfModule mod_weblogic.c>

WebLogicCluster<managed_server1_hostname/IP_address>,<managed_server2_hostna
me/IP_address>
    DynamicServerList OFF
    Debug ON
    IdemPotent OFF
</IfModule>
<Location /yantra>
    SetHandler weblogic-handler
</Location>
```

5. Create security or SSL certificate, if necessary. If you do not have a CA-signed certificate, you can get one from the Certificate Authority

companies such as VeriSign. For more information about security or SSL certificates, see [Section 15.5, "Security Certificates"](#).

6. Restart Apache, and verify access with any browser.

Note: Continue and complete steps 7-10 if using an RCP application only.

7. Copy the security certificate to the
`<RCP_EXTENSIONS_FOLDER>/truststore` directory.
8. Build the RCP client.
9. Edit the `locations.ycfg` file and modify the protocol, server, and port attributes of the Config element. Ensure that these attributes point to the proxy.
10. Start the client.

4.4 Setting Up the Image Server

If fetching images for RCP-based PCAs, you must set up an image server. You can set up any server (such as Apache) as your image server.

To set up the image server:

1. Install a web server on any system on which you intend to host the images. For example, you can install an Apache web server on a Windows system.
2. Use the default port # 80 (or any available port #) while installing the Image Server & exclude this port from the OS firewall, if required.
3. Store the images in any convenient location under the
`<IMAGE_SERVER_HOME>` directory. For example, you can store the images under the following directory:

`<IMAGE_SERVER_HOME>/icons/rcp`

where `<IMAGE_SERVER_HOME>` refers to the name of the directory to which the web server that you have installed points.

For more information about configuring connection settings to fetch images from the server, see [Chapter 15.4.1, "Configuring Connection Settings for Fetching Images from the Server"](#).

For example, if you install Apache as the web server, then to configure it as the image server, do the following:

- Edit the `httpd.conf` file to define an alias directive. You can find this file under the following directory structure:

```
<APACHE_HOME>/conf/httpd.conf
```

where `<APACHE_HOME>` refers to the name of the directory where you have installed Apache.

The following is a sample entry from the `httpd.conf` file:

```
Alias /icons ""  
  
<Directory ""  
    AllowOverride None  
    Order allow, deny  
    Allow from all  
</Directory>
```

where `/icons` is the `<virtual dir path>` that points to the `icons` directory.

For more information about how to define alias directives, go to http://httpd.apache.org/docs/2.2/en/mod/mod_alias.html#alias. This link provides information about alias directives for Apache version 2.2.

- Add a new entry or edit the existing entry for configuring the port. For example, add a new entry: `Listen 80` in the `httpd.conf` file. This sets up the server to listen to port number 80 (default setting).
- Restart the web server.

When we apply the above configuration, the URL:

`http://<IMAGE_SERVER_HOST_NAME>:<port>/icons` points to the local directory and the contents in the local directory are served by the web server.

Note: Test to ensure that the images are accessible through the browser. For example, `http://<IMAGE_SERVER_HOST_NAME>:<port>/<virtual dir path>/rcp/<IMAGE_FILE_NAME>` from any system. If the images are not displayed, the image server is not configured properly.

Installing and Configuring Database Tier Software on UNIX/Linux

This chapter describes how to install and configure the database tier software to run the Sterling Multi-Channel Fulfillment Solution in a Windows environment.

This chapter also provides the information required to complete [Step 5](#) and [Step 6](#) indicated on the [Installation Checklist](#) on page 1.

Before installing your database server, verify that you have the applicable software versions. For more information see [Chapter 2, "System Requirements"](#).

5.1 Database Sizing

Database sizing is designed to give you estimates of the database growth and to assist in planning the disk requirements. The planning of the capacity required in your company and the steps to estimate the disk size are described in [Section 5.1.1, "Capacity Planning"](#) on page 41, [Section 5.1.2, "Disk Estimation for the Distributed Order Management Module"](#) on page 42 and [Section 5.1.3, "Disk Estimation for the Networked Warehouse Management System Module"](#) on page 45.

5.1.1 Capacity Planning

There are many factors to consider when estimating the amount of disk space that is required for the Sterling Multi-Channel Fulfillment Solution. As a result, trying to consider all growth factors is impractical because the user may not know the answers to many questions that are required to do a detailed forecast. Over the years the cost of disks has dramatically decreased, and the capacity and speed of disks has

increased. The method of how information system managers order disk capacity has also changed from purchasing disk arrays that are dedicated to a particular database server and project to the concept of SANS.

The Sterling Multi-Channel Fulfillment Solution provides a methodology to estimate your initial disk requirements. Consider the confidence that you have in your data estimates when making the final purchase decision and adjust accordingly. After the initial purchase and production deployment, disk growth should be tracked for future purchase forecasts.

- If you use or are planning to use the Distributed Order Management (DOM) module, use [Table 5–1, "Steps for Disk Space Estimation for the Order Management Module"](#) on page 44.
- If you use or are planning to use both the Distributed Order Management (DOM) and the networked Warehouse Management System (nWMS) modules, use [Table 5–1, "Steps for Disk Space Estimation for the Order Management Module"](#) on page 44 and [Table 5–2, "Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS"](#) on page 47.
- If you are planning to use only the nWMS module use [Table 5–3, "Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS"](#) on page 49.

5.1.2 Disk Estimation for the Distributed Order Management Module

The disk estimation provided here pertains to the Order Management module of the Sterling Multi-Channel Fulfillment Solution.

The estimation methodology consists of three parts:

1. Estimate the number of orders and order lines you expect to keep in the database.
2. Multiply the number obtained in [Step 1](#) mentioned above, by a storage usage factor.
3. Finally add a minimum base amount.

However, the following information is essential to keep in mind before calculating the estimated disk space:

Note 1

You need to gather some information about the amount of time required to maintain the database, such as:

1. How long do you plan to keep data in the main transactional database before orders are purged to the history database?
2. How long are orders kept in the history database before they are purged?
3. Are you purchasing the storage for the first few years into the implementation?

Consider the following examples to achieve answers for the above mentioned questions.

Case 1 You need to purchase storage for the first 3 years of the implementation, and your company's data retention policy says that you have to keep data online in the main transactional database for 1 year and in the history database for another 5 years. Orders that are older than 6 years are purged from the system.

The following solution lets you achieve this goal:

If you need to purchase storage to cover the first 3 years of implementation, that storage has to be sufficient for 3 years worth of data. At the end of year 3, your database has the data for the third year in the main transactional database while the data for the first and second years is in the history. In this example, you should enter the number 3 as the number of years worth of orders that you expect to keep in the database.

Case 2 The Sterling Multi-Channel Fulfillment Solution has been in production for 10 years and your company's data retention policy says that you have to keep data online in the main transactional database for 1 year and in the history database for another 5 years. Orders that are older than 6 years are purged from the system. Given the same data retention policy as above, how much storage is required?

At the end of the tenth year, the database has the data for the tenth year in the main transactional database and the data for the fifth, sixth, seventh, eighth and ninth years in the history. Therefore, the database has six years (as dictated by the data retention policy) in

the database. In this example, you should enter the number 6 as the number of years worth of orders that you expect to be kept in the database.

Note 2

The order discussed in [Table 5–1, "Steps for Disk Space Estimation for the Order Management Module"](#) on page 44 includes sales, transfer, return, and work orders.

Note 3

This storage estimate is for work-in-progress tables that are used as part of order processing. When the orders are processed, the records in these tables can be purged from the system. These tables include the YFS_IMPORT, YFS_EXPORT, and so forth. ***You are strongly urged to aggressively purge data from these tables.***

Note 4

When procuring your storage, ensure that the storage device has at least the amount of usable space specified in [Step 8 of Table 5–1, "Steps for Disk Space Estimation for the Order Management Module"](#). This table provides an idea of the usable space for the storage device in your company. However, the actual amount you may need to order, is a factor of Redundant Array of Inexpensive Disks (RAID) set up. This disk subsystem is composed of more than one disk drive to provide improved reliability, response time, and storage capacity.

Now that you have noted the above points you can proceed to the estimation of required disk space as outlined in [Table 5–1](#).

Table 5–1 Steps for Disk Space Estimation for the Order Management Module

- | | | |
|----|---|-------|
| 1. | Enter the number of years worth of information to be kept in the system (retention time). For a more detailed example, refer to "Note 1". | _____ |
| 2. | Enter the number of orders you expect to be in the system during the time period specified in Step 1 . For the different types of orders refer to "Note 2". | _____ |
| 3. | Enter the number of order lines present in a typical order. | _____ |

Table 5–1 Steps for Disk Space Estimation for the Order Management Module

4.	Enter the number of order lines that are to be stored in the database (multiply the values provided in Step 2 and Step 3).	_____
5.	Enter the order line multiplier: Choose one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system: (a) 30 KB - This is primarily used for order management with very little customization. (b) 35 KB - This is primarily used for order management with moderate amount of customization.	_____
6.	Multiply the expected number of order lines from Step 4 and the storage factor from Step 5 .	_____
7.	The minimum base storage requirement.	150 MB
8.	The minimum operational storage requirements for the Sterling Multi-Channel Fulfillment Solution. For more information on the storage estimate, refer to " Note 3 ".	500 MB
9.	Enter the total estimated storage obtained by adding the values from Step 6 , Step 7 , and Step 8 . For more information on the amount of usable space, refer to " Note 4 ".	_____

5.1.3 Disk Estimation for the Networked Warehouse Management System Module

The disk estimation discussed in this section pertains to the networked WMS module of the Sterling Multi-Channel Fulfillment Solution.

This estimation methodology consists of three parts:

- 1. Estimate the number of shipment lines you expect to keep in the database.
- 2. Multiply the number obtained in [Step 1](#) by a storage usage factor depending on the specifics of your implementation.
- 3. Add a minimum base amount for each warehouse or stockroom that you have defined.

If you are planning to use both the Sterling Multi-Channel Fulfillment Solution DOM and nWMS modules use [Table 5–2, "Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS"](#) on page 47 or else if you are planning to use only the nWMS

module use [Table 5–3, "Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS"](#) on page 49.

However, the following information is essential to keep in mind before calculating the estimated disk space:

Note 1

You need to gather some information about the amount of time required to maintain the database, such as:

1. How long do you plan to keep data in the main transactional database before shipment data is purged to the history database?
2. How long is the shipment data kept in the history database before it is purged?
3. Are you purchasing the storage for the first few years into the implementation?

Consider the following example to achieve answers for the above mentioned questions.

Case 1 You need to purchase storage for the first 2 years of the implementation, and your company's data retention policy says that you have to keep data online in the main transactional database for 1 year and in the history database for another year. Shipments that are older than 2 years are purged from the system.

The following solution lets you achieve this goal:

If you need to purchase storage to cover the first 2 years of implementation, that storage has to be sufficient for 2 years worth of data. At the end of year 2, your database has data from the second year in the main transactional database while the data from the first year is in the history. In this example, you should enter the number 2 as the number of years worth of shipment-related data that you expect to keep in the database.

Note 2

The shipment lines discussed in [Table 5–2](#) on page 47 and [Table 5–3](#) on page 49 include space requirements for demand-based replenishment.

Note 3

This storage estimate is for work-in-progress tables that are used as part of the shipment and receipt processing. When the shipments are processed, the records in these tables can be purged from the system. These tables include the YFS_IMPORT, YFS_EXPORT, YFS_TASK, YFS_TASK_STATUS_AUDIT, and so forth. ***You are strongly urged to aggressively purge data from these tables.***

Note 4

When procuring your storage, ensure that the storage device has at least the amount of usable space specified in the last step of [Table 5–2](#) on page 47 or [Table 5–3](#) on page 49. These tables provide an idea of the usable space for the storage device in your company. However, the actual amount you may need is a factor of Redundant Array of Inexpensive Disks (RAID) set up. This disk subsystem is composed of more than one disk drive to provide improved reliability, response time and storage capacity.

Now that you have noted the above points you can proceed to the estimation of required disk space as outlined in [Table 5–2](#) on page 47 or [Table 5–3](#) on page 49.

Table 5–2 Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS

-
- | | | |
|----|---|-------|
| 1. | Enter the number of years worth of information to be kept in the system (retention time). For a more detailed example, refer to "Note 1". | _____ |
| 2. | Enter the number of shipment lines you expect to be in the system during the time period specified in Step 1 . For the different types of shipments, refer to "Note 2". | _____ |

Table 5–2 Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS

3. Enter the shipment line multiplier. This includes demand-based replenishment. Choose one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system: _____
 - (a) 10 KB - for warehouses using no tag-controlled items, no serial tracking, no LNPNS or only Pallet LPNs, and more than 80% PARCEL shipping.
 - (b) 12 KB - for warehouses using no tag-controlled items, no serial tracking, no LNPNS or only Pallet LPNs, and PARCEL as well as TL - LTL shipping.
 - (c) 15 KB - for warehouses using tag-controlled items, no serial tracking, no LNPNS or only Pallet LPNs, and more than 80% TL - LTL shipping.
 - (d) 20 KB - for warehouses using no tag-controlled items, no serial tracking, largely CASE LPNs, and more than 80% PARCEL shipping.
 - (e) 25 KB - for warehouses using tag-controlled items, serial tracking, largely CASE LPNs, and PARCEL as well as TL - LTL shipping.
 - (f) 30 KB - for warehouses using tag-controlled items, serial tracking, largely CASE LPNs, and more than 80% TL - LTL shipping.
4. Multiply the expected number of shipment lines from [Step 2](#) and the storage factor from [Step 3](#). _____
5. Enter the number of receipt lines you expect to be in the system during the time period specified in [Step 1](#). _____
6. Enter the receipt line multiplier. Choose one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system: _____
 - (a) 25 KB - for warehouses using no tag-controlled items, no serial tracking, and no LNPNS.
 - (b) 27 KB - for warehouses using no tag-controlled items, no serial tracking, and no LNPNS or only Pallet LPNs.
 - (c) 35 KB - for warehouses using no tag-controlled items, no serial tracking, and more than 80% CASE LPNs.
 - (d) 40 KB - for warehouses using tag-controlled items, serial tracking, and more than 80% CASE LPNs.

Table 5–2 Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS

7.	Multiply the expected number of receipt lines from Step 5 and the storage factor from Step 6 .	_____
8.	Enter the number of warehouses planned:	
	(a) Enter the number of stores or stock rooms planned.	_____
	(b) Enter the number of other warehouses planned.	_____
9.	Calculate the minimum space required for your set up based on the data in Step 8 and the minimum storage requirement given below:	
	(a) 20 MB for each store or stock room.	_____
	(b) 50 MB for each other warehouse.	_____
10.	The minimum operational storage requirements for the Sterling Multi-Channel Fulfillment Solution. For more information on the storage estimates refer to the " Note 3 ".	500 MB
11.	Enter the total estimated storage obtained by adding the values from Step 4 , Step 7 , Step 8 , Step 9 and Step 10 . For more information on the amount of usable space, refer to " Note 4 ".	_____
12.	Enter the value of Step 9 from Table 5–1 on page 44.	_____
13.	Enter the total estimated storage obtained by adding the values from Step 11 and Step 12 . For more information on the amount of usable space, refer to " Note 4 ".	_____

Table 5–3 Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS

1.	Enter the number of years worth of information to be kept in the system (retention time). For a more detailed example, refer to " Note 1 ".	_____
2.	Enter the number of shipment lines you expect to be in the system during the time period specified in Step 1 . For the different types of shipments, refer to " Note 2 ".	_____

Table 5–3 Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS

3. Enter the shipment line multiplier. This factor includes demand-based replenishment. Choose from one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system:

(a) 25 KB - for warehouses using no tag-controlled items, no serial tracking, no LNPNS or only Pallet LPNs, and more than 80% PARCEL shipping.

(b) 27 KB - for warehouses using no tag-controlled items, no serial tracking, no LNPNS or only Pallet LPNs, and PARCEL as well as TL - LTL shipping.

(c) 30 KB - for warehouses using tag-controlled items, no serial tracking, no LNPNS or only Pallet LPNs, and more than 80% TL - LTL shipping.

(d) 35 KB - for warehouses using no tag-controlled items, no serial tracking, largely CASE LPNs, and more than 80% PARCEL shipping.

(e) 40 KB - for warehouses using tag-controlled items, serial tracking, largely CASE LPNs, and PARCEL as well as TL - LTL shipping.

(f) 50 KB - for warehouses using tag-controlled items, serial tracking, largely CASE LPNs, and more than 80% TL - LTL shipping.
4. Multiply the expected number of shipment lines from [Step 2](#) and the storage factor from [Step 3](#).
5. Enter the number of receipt lines you expect to be in the system during the time period specified in [Step 1](#).
6. Enter the receipt line multiplier. Choose from one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system:

(a) 25 KB - for warehouses using no tag-controlled items, no serial tracking, and no LNPNS.

(b) 27 KB - for warehouses using no tag-controlled items, no serial tracking, and no LNPNS or only Pallet LPNs.

(c) 35 KB - for warehouses using no tag-controlled items, no serial tracking, and more than 80% CASE LPNs.

(d) 40 KB - for warehouses using tag-controlled items, serial tracking, and more than 80% CASE LPNs.

Table 5–3 Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS

7.	Multiply the expected number of receipt lines from Step 5 and the storage factor from Step 6 .	_____
8.	Enter the number of warehouses planned:	
	(a) Enter the number of stores or stock rooms planned.	_____
	(b) Enter the number of other warehouses planned.	_____
9.	Calculate the minimum space required for your set up based on the data in Step 8 and the minimum storage requirement given below:	
	(a) 20 MB for each store or stock room.	_____
	(b) 50 MB for each other warehouse.	_____
10.	The minimum operational storage requirements for the Sterling Multi-Channel Fulfillment Solution. For more information on the storage estimates refer to " Note 3 ".	500 MB
11.	Enter the total estimated storage obtained by adding the values from Step 4 , Step 7 , Step 8 , Step 9 and Step 10 . For more information on the amount of usable space, refer to " Note 4 ".	_____

5.1.4 Tracking and Estimating Future Disk Requirements

You should track your actual database storage usage and the number of database records regularly. Correlating these two metrics enabled you to plan your future disk requirements. Moreover, determining the average amount of space used for each order line or shipment line, enables you to accurately predict your future growth requirements.

5.2 Installing Oracle (UNIX/Linux)

You can use an Oracle database for maintaining information on the Sterling Multi-Channel Fulfillment Solution. The following sections provide the necessary steps to install and configure an Oracle database for production.

To install Oracle:

Follow the steps below to install Oracle with single or multiple byte characters:

1. If you do not have Oracle installed, follow the installation procedures in your Oracle Installation manuals.
2. Run the create instance procedure. Use a character set appropriate for your desired language.

```
CHARACTER SET "UTF8"
```

3. Configure the INIT<INSTANCE_NAME>.ORA file for Oracle as follows:

```
open_cursors= <set to appropriate value>
```

For example, the minimum value for WebLogic equals number of threads (across all application servers) + (connection pool size X prepared statement pool size)

```
cursor_sharing=similar  
compatible=<10.2.0.3>  
timed_statistics=true  
db_block_size=8192  
optimizer_mode=CHOOSE
```

If you are using multi-byte character set, set the following and restart Oracle:

```
nls_length_semantics=CHAR
```

Alternatively you can run:

```
alter session set nls_length_semantics = CHAR
```

prior to running any create table scripts.

Setting this attribute ensures that the field sizes are not impacted by the number of bytes a data type can store. For example, Varchar(40) would now be able to store 40 Japanese characters instead of 40/3 bytes in the UTF-8 character set.

Note: When you change the multi-byte character set to CHAR by setting `nls_length_semantics = CHAR`, Oracle reserves space equivalent to 'n' chars, which is more than 'n' bytes. Therefore, when you run the `dbverify.sh` command, the reduced entries in table columns are printed in the `EFrame_Drops.lst` file.

4. Download the Oracle JDBC driver `ojdbc14.jar` from the Oracle website and copy it to a well known location for reference during installation.

The Oracle JDBC driver can be found at:

http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/index.html

Download the `ojdbc14.jar` file for Oracle 10.2.0.3.

5.2.1 Oracle Database User Privileges

Unless specifically stated for a given task, the Sterling Multi-Channel Fulfillment Solution user does not require database administrator (DBA) privileges.

Following are some of the basic privileges that should be granted to the Sterling Multi-Channel Fulfillment Solution administrative user who is involved in creating and modifying the Oracle database:

- ALTER ANY SEQUENCE
- ALTER SESSION
- ALTER USER DEFAULT ROLE
- CREATE ANY SEQUENCE
- CREATE PROCEDURE
- CREATE SEQUENCE
- CREATE SESSION
- CREATE SYNONYM
- CREATE TABLE
- CREATE TRIGGER

- CREATE TYPE
- CREATE VIEW
- EXECUTE ANY PROCEDURE
- EXECUTE ANY TYPE
- GRANT "CONNECT"
- INSERT ANY TABLE
- SELECT ANY DICTIONARY
- SELECT ANY SEQUENCE
- SELECT ANY TABLE
- SELECT_CATALOG_ROLE
- UPDATE ANY TABLE

The following are some of the basic privileges that should be granted to the application user who will only be running the application:

- CREATE SESSION
- ALTER SESSION
- EXECUTE ANY PROCEDURE
- INSERT ANY TABLE
- UPDATE ANY TABLE
- SELECT ANY TABLE
- SELECT ANY SEQUENCE

Note: Ensure that the user who is responsible for creating and modifying the Oracle database has a specified quota (extent) assigned to him in the tablespace even if the user was assigned a unlimited tablespace when the user was created. Otherwise, the installer will throw the "ORA-01950: no privileges on tablespace name" error.

Note: If you are using text indexes, you must also have privileges for CTXAPP or CTXCAT, depending on the type of text indexes you are using.

5.2.2 Configuring an Oracle Database for Production

You need to configure your Oracle database for running in a production environment with the Sterling Multi-Channel Fulfillment Solution. To configure an Oracle database for a production environment, you must:

- [Size the database](#) by estimating the required disk space.
- Create views and db_link or synonyms for integrating with the Sterling Warehouse Management System installation.
- Set the [database connection](#) properties.

To create the Oracle database to handle multiple byte characters:

1. Do not modify the Sterling Multi-Channel Fulfillment Solution DDL.
2. Choose the correct data encoding format for your language. See "[To install Oracle:](#)" on page 51 for more information.
3. Choose the character set suitable for your language. See "[To install Oracle:](#)" on page 51 for specific settings to ensure the database field sizes.

To set up scripts (if you are using locally managed tablespaces or another utility to size your database):

To configure your Oracle database for your production environment, you must set up and run a series of scripts to create the tables, indexes, sequences, and so forth for your schema.

These script files reside in the `<INSTALL_DIR>/database/oracle/scripts/` directory. The `yfs_master_db_script.sql` script is the master script that calls all view scripts required for creating views. Table, index, and sequence creation DDLs are created during installation. These reside in `<INSTALL_DIR>/repository/scripts`.

To set up the scripts:

1. Create tablespaces where the Sterling Multi-Channel Fulfillment Solution tables and indexes reside.
2. Only complete this step if you are manually creating database tables after installation (instead of having installation create them automatically): modify the `<INSTALL_DIR>/repository/scripts/EFrame_TableChanges.sql` file to reference your newly created tablespaces.

The DDLs in the Sterling Multi-Channel Fulfillment Solution scripts create a standard set of indexes. You may need to create additional indexes or modify existing indexes according to your business practice.

Note: Index creations should be run only *after* everything else in the `tablechanges` script.

To run the scripts:

1. Log into the Oracle server manager as `sysdba`.
2. Create the user that is the designated schema owner.
3. Grant the privileges listed in [Section 5.2.1, "Oracle Database User Privileges"](#) on page 53 to the newly created user.
4. Log out of the Oracle Server Manager and log back in as the newly created user.
5. Verify the database as described in [Section 13.1.1.2, "Verifying the Database Schema"](#) on page 160.
6. Load the Sterling Multi-Channel Fulfillment Solution database factory defaults as described in [Section 13.1.1.3, "Loading the Sterling Multi-Channel Fulfillment Solution Database Factory Defaults After Installation"](#) on page 165.

7. Check for the degree of parallelism, using information from the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

5.2.2.1 Enabling the Text Search Feature

The Sterling Multi-Channel Fulfillment Solution supports two types of text search indexes on Oracle databases: CTXCAT and CONTEXT. The CTXCAT index supports automatic updating of text search indexes, whereas, the CONTEXT index does not support automatic updating of text search indexes. Sterling Commerce recommends that you use the CTXCAT index.

For information on how to create the text search indexes, refer to the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.

This section explains the following:

- [Enabling the Text Search Feature for CTXCAT Index](#)
- [Enabling the Text Search Feature for CONTEXT Index](#)

5.2.2.1.1 Enabling the Text Search Feature for CTXCAT Index

The CTXCAT index automatically updates text search indexes. Therefore, the DBA need not manually run the `EFrame_TextIndexUpdates.sql` script to update text search indexes.

To enable the text search feature on an Oracle database using the CTXCAT index:

1. Make sure that the Oracle database is configured with the Oracle Text feature.
2. Log in to the Oracle server with a user ID having the CTXAPP privilege.
3. Verify that the text search index creation was successful.
4. Edit the `customer_overrides.properties` file that is located in the `<INSTALL_DIR>/properties` directory to add the following entries:

```
yfs.yfs.db.textsearch=Y
yfs.yfs.db.textsearch.oracle.contexttype=ctxcat
```

5.2.2.1.2 Enabling the Text Search Feature for CONTEXT Index

The CONTEXT index does not automatically update text search indexes. Therefore, the DBA has to manually update text search indexes by running the `EFrame_TextIndexUpdates.sql` script.

To enable the text search feature on Oracle database using the CONTEXT index:

1. Make sure that the Oracle database is configured with the Oracle Text feature.
2. Log in to the Oracle server with a user ID having the CTXAPP privilege.

Note: The CONTEXT type text search indexes that are created on Oracle database using the `EFrame_TextIndexAdds.sql` script are not updated automatically. The DBA has to run the `EFrame_TextIndexUpdates.sql` script to update the CONTEXT type text search indexes whenever required using scheduled jobs. The frequency of these scheduled jobs can be decided by the DBA.

3. Verify that the text search index creation was successful.
4. Edit the `customer_overrides.properties` file that is located in the `<INSTALL_DIR>/properties/` directory to add the following entries:

```
yfs.yfs.db.textsearch=Y  
yfs.yfs.db.textsearch.oracle.contexttype=context
```

5.2.3 Using an Oracle Database Server

You can use an Oracle 10g database with the Sterling Multi-Channel Fulfillment Solution. See [Chapter 2, "System Requirements"](#) for supported version information. The Sterling Multi-Channel Fulfillment Solution supports Oracle 10g in a single node database environment.

To use an Oracle 10g database:

- Create the database. Refer to the Oracle documentation for information about creating the database, including creating a schema repository, login, and tablespace. Be sure to install the correct version and patches.

- Configure the database by completing the following tasks:
 - Setting Database Parameters in Oracle
 - Rolling Back or Undoing Changes in Oracle
 - Granting Permissions in Oracle
 - Installing the JDBC Driver in Oracle

5.2.3.1 Setting Database Parameters in Oracle

For information about required parameter settings in your Oracle database, see the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

5.2.3.2 Rolling Back or Undoing Changes in Oracle

You can roll back or undo changes in Oracle using the following method:

(Oracle versions 10g or later) These versions support AUTO UNDO management. It is recommended that you use this option. This avoids manual monitoring of UNDO segments.

If a server is upgraded from Oracle 9i, set the UNDO_MANAGEMENT=AUTO parameter in init<SID>.ora. Your database administrator needs to determine the UNDO_RETENTION setting. Ensure that the file system which has the UNDOTBS1 tablespace has enough space to use the AUTOGROW setting.

5.2.4 Installing the JDBC Driver in Oracle

The Sterling Multi-Channel Fulfillment Solution requires the appropriate JDBC driver for Oracle 10g databases. These drivers are thin client, 100% Pure Java™ JDBC drivers. See [Chapter 2, "System Requirements"](#) for supported version information.

After you obtain the correct JDBC driver file, record the absolute path to its location on your system. You must supply this absolute path when you install the Sterling Multi-Channel Fulfillment Solution.

5.3 Installing DB2 (UNIX/Linux)

You can use a DB2 database for maintaining information on the Sterling Multi-Channel Fulfillment Solution. The following sections provide the necessary steps to install and configure a DB2 database for production.

To install DB2:

1. If you do not have DB2 installed, follow the installation procedures in your DB2 Installation manual.

Note: When creating the DB2 database, the appropriate codepage needs to be selected for international language characters (for example, UTF-8).

2. You need to set the following parameter to avoid memory leaks and DB2 crashes:

```
db2set DB2_NUM_CKPW_DAEMONS=0
```

5.3.1 DB2 Database User Privileges

The DBADM role is required for performing administrative operations in the DB2 database.

5.3.2 Configuring a DB2 Database for Production

You need to configure your DB2 database for running in a production environment with the Sterling Multi-Channel Fulfillment Solution. To configure a DB2 database for a production environment, you must:

- [Size the database](#) by estimating the required disk space.
- Set the [database connection](#) properties.

Note: The installation script creates tables and indexes. Certain tables require a page size of 32K. You should have a tablespace to accommodate such tables. DB2 automatically places tables and indexes in the available tablespaces using its internal logic. You can move the tables to a different tablespace after the installation is complete.

Manually creating objects on DB2

To set up the scripts:

1. Create tablespaces where the Sterling Multi-Channel Fulfillment Solution tables and indexes reside.
2. Only complete this step if you are manually creating database tables after installation (instead of having installation create them automatically): modify the
`<INSTALL_DIR>/repository/scripts/EFrame_TableChanges.sql`
file to reference your newly created tablespaces.

The DDLs in the Sterling Multi-Channel Fulfillment Solution scripts create a standard set of indexes. You may need to create additional indexes or modify existing indexes according to your business practice.

To run the scripts:

1. Log into the DB2 server manager as the database administrator.
2. Create the user that is the designated schema owner.
3. Grant the privileges listed in [Section 5.3.1, "DB2 Database User Privileges"](#) on page 60 to the newly created user.
4. Log out of the DB2 Server Manager and log back in as the newly created user.
5. Verify the database as described in [Section 13.1.1.2, "Verifying the Database Schema"](#) on page 160.
6. Load the Sterling Multi-Channel Fulfillment Solution database factory defaults as described in [Section 13.1.1.3, "Loading the Sterling Multi-Channel Fulfillment Solution Database Factory Defaults After Installation"](#) on page 165.

7. Check for the degree of parallelism, using information from the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

Manually creating views on DB2

To configure your DB2 database for your production environment, you must set up and run a series of scripts to create the tables, indexes, sequences, and so forth for your schema.

These script files reside in the <INSTALL_DIR>/database/db2/scripts/ directory.

This is the list of scripts to be edited using a SQL tool:

- ImportExport_View.sql
- Interop_Views.sql
- InvSnapshot_vw.sql
- yfs_cross_reference_vw.sql
- yfs_iba_ord_demand_vw.sql
- yfs_iba_resv_demand_vw.sql
- yfs_invtmdmddl_vw.sql
- yfs_order_release_line.sql
- yfs_order_release_line_vw.sql

5.3.2.1 Enabling the Text Search Feature

To enable the text search feature on DB2 database:

1. Make sure that the DB2 database is configured with the Net Search Extender plug-in.
2. Log in to the DB2 server using the Command Editor or Command Line Processor with a user ID having DBA privileges.

Note: The text search indexes that are created on DB2 database using the `<INSTALL_DIR>/repository/scripts/EFrame_TextIndexAdds.sql` script are automatically updated every 6 hours. The DBA can modify this script to change this frequency, if necessary. Before running the `EFrame_TextIndexAdds.sql` script, the DBA must update the `"/*Database*/"` string in the `EFrame_TextIndexAdds.sql` script and specify the database name.

3. Verify that the text search index creation was successful.
4. Edit the `customer_overrides.properties` file that is located in the `<INSTALL_DIR>/properties/` directory to add the following entry:
`yfs.yfs.db.textsearch=Y`

For information on how to create the text search indexes, see the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.

5.3.3 Using a DB2 Database Server

You can use a DB2 database with the Sterling Multi-Channel Fulfillment Solution. See [Chapter 2, "System Requirements"](#) for supported version information.

To use a DB2 server:

- Create the database. Refer to the DB2 documentation for information about creating the database, including creating a schema repository, login, and tablespace. Be sure to install the correct version and patch. Be sure to install the client components and compilers before you install the fixpack.
- Configure the database by completing the following tasks:
 - Installing Client Components, Compilers, and Fix Pack
 - Setting Parameters for DB2
 - Granting Permissions for DB2
 - Installing JDBC Drivers for DB2

5.3.3.1 Installing Client Components, Compilers, and Fix Pack

The Sterling Multi-Channel Fulfillment Solution uses stored procedures for DB2. You must install or set up the following components:

1. Install the Administration client.
2. Install the Sterling Multi-Channel Fulfillment Solution Development clients.
3. Install the necessary fix pack after you install the client components and compilers. Otherwise, the clients will overwrite the fix pack binaries.
4. Set the path for the compiler by using the **db2set** command.

For more information about these tasks, see the IBM documentation.

5.3.3.2 Setting Parameters for DB2

For information about required parameter settings in your DB2 database, see the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

5.3.3.3 Installing JDBC Drivers for DB2

For DB2, install the appropriate DB2 JDBC Type 4 driver and any correlating patches. See [Chapter 2, "System Requirements"](#) for supported version information.

You can obtain these files from the IBM Web site. After you obtain this JDBC driver, record the absolute path to its location on your system. You must supply this absolute path during installation.

If the JDBC driver provided by your database vendor is distributed among multiple files, you must place all the files that comprise the JDBC driver into one .jar file. Follow these steps to create one .jar file:

1. Identify all the vendor database jar files for the JDBC driver.
2. Create a temporary working directory (mkdir wd; cd wd).
3. Extract the contents of each file used for the JDBC driver using the jar utility into the temporary working directory (jar xvf <jdbc.jar> for each supplied jar file).

Note: Various Sterling Multi-Channel Fulfillment Solution scripts, such as the one used for loading the factory defaults, specify a `DB_DRIVER`. The `DB_DRIVER` specified must include ***all*** of these JAR files. The `DB_DRIVER` setting is located in `sandbox.cfg`. To make changes to the `DB_DRIVER` setting, edit and save the file, then run `setupfiles.sh`.

4. Bundle the files in the temporary working directory into one file using the `jar` utility (`jar cvf new.jar *`).
5. Record the absolute path to the `.jar` file you created on the Preinstallation Checklist.

The type-4 driver does not require a separate Java listener running on the database server. Instead, connect directly to the DB2 port.

Installing and Configuring Database Tier Software on Windows

This chapter describes how to install and configure the database tier software to run the Sterling Multi-Channel Fulfillment Solution in a Windows environment.

This chapter also provides the information required to complete [Step 5](#) and [Step 6](#) indicated on the [Installation Checklist](#) on page 1.

Before installing your database server, verify that you have the applicable software versions. For more information see [Chapter 2, "System Requirements"](#).

6.1 Creating and Configuring the Database Server (Windows)

You must install, create, and configure a database so that each Sterling Multi-Channel Fulfillment Solution instance has a dedicated schema and login for the database.

Caution: If you are reinstalling the Sterling Multi-Channel Fulfillment Solution, be aware that data in your existing database will be deleted. To prevent this, either back up the existing database or save it under a different name.

After creating and configuring your database, recycle the database. Then stop and restart the Sterling Multi-Channel Fulfillment Solution to apply the changes.

In a Windows environment, the Sterling Multi-Channel Fulfillment Solution supports the following databases:

- MS SQL 2000
- MS SQL 2005
- Oracle® 10g

See [Chapter 2, "System Requirements"](#) for supported version information.

6.2 Database Sizing

Database sizing is designed to give you estimates of the database growth and to assist in planning the disk requirements. The planning of the capacity required in your company and the steps to estimate the disk size are described in [Section 6.2.1, "Capacity Planning"](#) on page 68, [Section 6.2.2, "Disk Estimation for the Distributed Order Management Module"](#) on page 69 and [Section 6.2.3, "Disk Estimation for the Networked Warehouse Management System Module"](#) on page 72.

6.2.1 Capacity Planning

There are many factors to consider when estimating the amount of disk space that is required for the Sterling Multi-Channel Fulfillment Solution. As a result, trying to consider all growth factors is impractical because the user may not know the answers to many questions that are required to do a detailed forecast. Over the years the cost of disks has dramatically decreased, and the capacity and speed of disks has increased. The method of how information system managers order disk capacity has also changed from purchasing disk arrays that are dedicated to a particular database server and project to the concept of SANs.

The Sterling Multi-Channel Fulfillment Solution provides a methodology to estimate your initial disk requirements. Consider the confidence that you have in your data estimates when making the final purchase decision and adjust accordingly. After the initial purchase and production deployment, disk growth should be tracked for future purchase forecasts.

- If you use or are planning to use the Distributed Order Management (DOM) module, use [Table 6–1, "Steps for Disk Space Estimation for the Order Management Module"](#) on page 71.
- If you use or are planning to use both Distributed Order Management (DOM) and networked Warehouse Management System (nWMS) modules, please use [Table 6–1, "Steps for Disk Space Estimation for the Order Management Module"](#) on page 71 and [Table 6–2, "Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS"](#) on page 74.
- If you are planning to use only the nWMS module use [Table 6–3, "Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS"](#) on page 76.

6.2.2 Disk Estimation for the Distributed Order Management Module

The disk estimation provided here pertains to the Order Management module of the Sterling Multi-Channel Fulfillment Solution.

The estimation methodology consists of three parts:

1. Estimate the number of orders and order lines you expect to keep in the database.
2. Multiply the number obtained in [Step 1](#) mentioned above, by a storage usage factor.
3. Finally add a minimum base amount.

However, the following information is essential to keep in mind before calculating the estimated disk space:

Note 1

You need to gather some information about the amount of time required to maintain the database, such as:

1. How long do you plan to keep data in the main transactional database before orders are purged to the history database?
2. How long are orders kept in the history database before they are purged?
3. Are you purchasing the storage for the first few years into the implementation?

Consider the following examples to achieve answers for the above mentioned questions.

Case 1 You need to purchase storage for the first 3 years of the implementation, and your company's data retention policy says that you have to keep data online in the main transactional database for 1 year and in the history database for another 5 years. Orders that are older than 6 years are purged from the system.

The following solution lets you achieve this goal:

If you need to purchase storage to cover the first 3 years of implementation, that storage has to be sufficient for 3 years worth of data. At the end of year 3, your database has the data for the third year in the main transactional database while the data for the first and second years is in the history. In this example, you should enter the number 3 as the number of years worth of orders that you expect to keep in the database.

Case 2 The Sterling Multi-Channel Fulfillment Solution has been in production for 10 years and your company's data retention policy says that you have to keep data online in the main transactional database for 1 year and in the history database for another 5 years. Orders that are older than 6 years are purged from the system. Given the same data retention policy as above, how much storage is required?

At the end of the tenth year, the database has the data for the tenth year in the main transactional database and the data for the fifth, sixth, seventh, eighth and ninth years in the history. Therefore, the database has six years (as dictated by the data retention policy) in the database. In this example, you should enter the number 6 as the number of years worth of orders that you expect to be kept in the database.

Note 2

The order discussed in [Table 6–1, "Steps for Disk Space Estimation for the Order Management Module"](#) on page 71 includes sales, transfer, return, and work orders.

Note 3

This storage estimate is for work-in-progress tables that are used as part of order processing. When the orders are processed, the records in these tables can be purged from the system. These tables include the YFS_IMPORT, YFS_EXPORT, and so forth. ***You are strongly urged to aggressively purge data from these tables.***

Note 4

When procuring your storage, ensure that the storage device has at least the amount of usable space specified in [Step 8 of Table 6–1, "Steps for Disk Space Estimation for the Order Management Module"](#). This table provides an idea of the usable space for the storage device in your company. However, the actual amount you might need to order, is a factor of Redundant Array of Inexpensive Disks (RAID) set up. This disk subsystem is composed of more than one disk drive to provide improved reliability, response time, and storage capacity.

Now that you have noted the above points you can proceed to the estimation of required disk space as outlined in [Table 6–1](#).

Table 6–1 Steps for Disk Space Estimation for the Order Management Module

-
- | | | |
|----|---|-------|
| 1. | Enter the number of years worth of information to be kept in the system (retention time). For a more detailed example, refer to "Note 1". | _____ |
| 2. | Enter the number of orders you expect to be in the system during the time period specified in Step 1 . For the different types of orders refer to "Note 2". | _____ |
| 3. | Enter the number of order lines present in a typical order. | _____ |
| 4. | Enter the number of order lines that are to be stored in the database (multiply the values provided in Step 2 and Step 3). | _____ |

Table 6–1 Steps for Disk Space Estimation for the Order Management Module

5.	Enter the order line multiplier: Choose one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system:	_____
	(a) 30 KB - This is primarily used for order management with very little customization.	
	(b) 35 KB - This is primarily used for order management with moderate amount of customization.	
6.	Multiply the expected number of order lines from Step 4 and the storage factor from Step 5 .	_____
7.	The minimum base storage requirement.	150 MB
8.	The minimum operational storage requirements for the Sterling Multi-Channel Fulfillment Solution. For more information on the storage estimate, refer to " Note 3 ".	500 MB
9.	Enter the total estimated storage obtained by adding the values from Step 6 , Step 7 , and Step 8 . For more information on the amount of usable space, refer to " Note 4 ".	_____

6.2.3 Disk Estimation for the Networked Warehouse Management System Module

The disk estimation discussed in this section pertains to the networked WMS module of the Sterling Multi-Channel Fulfillment Solution.

This estimation methodology consists of three parts:

- 1. Estimate the number of shipment lines you expect to keep in the database.
- 2. Multiply the number obtained in [Step 1](#) by a storage usage factor depending on the specifics of your implementation.
- 3. Add a minimum base amount for each warehouse or stockroom that you have defined.

If you are planning to use both the Sterling Multi-Channel Fulfillment Solution DOM and nWMS modules, use [Table 6–2, "Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS"](#) on page 74 or else if you are planning to use only the nWMS module use [Table 6–3, "Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS"](#) on page 76.

However, the following information is essential to keep in mind before calculating the estimated disk space:

Note 1

You need to gather some information about the amount of time required to maintain the database, such as:

1. How long do you plan to keep data in the main transactional database before shipment data is purged to the history database?
2. How long is the shipment data kept in the history database before it is purged?
3. Are you purchasing the storage for the first few years into the implementation?

Consider the following example to achieve answers for the above mentioned questions.

Case 1 You need to purchase storage for the first 2 years of the implementation, and your company's data retention policy says that you have to keep data online in the main transactional database for 1 year and in the history database for another year. Shipments that are older than 2 years are purged from the system.

The following solution lets you achieve this goal:

If you need to purchase storage to cover the first 2 years of implementation, that storage has to be sufficient for 2 years worth of data. At the end of year 2, your database has data from the second year in the main transactional database while the data from the first year is in the history. In this example, you should enter the number 2 as the number of years worth of shipment-related data that you expect to keep in the database.

Note 2

The shipment lines discussed in [Table 6–2](#) on page 74 and [Table 6–3](#) on page 76 include space requirements for demand-based replenishment.

Note 3

This storage estimate is for work-in-progress tables that are used as part of the shipment and receipt processing. When the shipments are processed, the records in these tables can be purged from the system.

These tables include the YFS_IMPORT, YFS_EXPORT, YFS_TASK, YFS_TASK_STATUS_AUDIT, and so forth. ***You are strongly urged to aggressively purge data from these tables.***

Note 4

When procuring your storage, ensure that the storage device has at least the amount of usable space specified in the last step of [Table 6–2](#) on page 74 and [Table 6–3](#) on page 76. These tables provide an idea of the usable space for the storage device in your company. However, the actual amount you might need is a factor of Redundant Array of Inexpensive Disks (RAID) set up. This disk subsystem is composed of more than one disk drive to provide improved reliability, response time and storage capacity.

Now that you have noted the above points you can proceed to the estimation of required disk space as outlined in [Table 6–2](#) on page 74 and [Table 6–3](#) on page 76.

Table 6–2 Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS

-
1. Enter the number of years worth of information to be kept in the system (retention time). For a more detailed example, refer to "Note 1". _____
 2. Enter the number of shipment lines you expect to be in the system during the time period specified in [Step 1](#). For the different types of shipments, refer to "Note 2". _____

Table 6–2 Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS

3. Enter the shipment line multiplier. This includes demand-based replenishment. Choose one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system: _____
 - (a) 10 KB - for warehouses using no tag-controlled items, no serial tracking, no LNPNs or only Pallet LPNs, and more than 80% PARCEL shipping.
 - (b) 12 KB - for warehouses using no tag-controlled items, no serial tracking, no LNPNs or only Pallet LPNs, and PARCEL as well as TL - LTL shipping.
 - (c) 15 KB - for warehouses using tag-controlled items, no serial tracking, no LNPNs or only Pallet LPNs, and more than 80% TL - LTL shipping.
 - (d) 20 KB - for warehouses using no tag-controlled items, no serial tracking, largely CASE LPNs, and more than 80% PARCEL shipping.
 - (e) 25 KB - for warehouses using tag-controlled items, serial tracking, largely CASE LPNs, and PARCEL as well as TL - LTL shipping.
 - (f) 30 KB - for warehouses using tag-controlled items, serial tracking, largely CASE LPNs, and more than 80% TL - LTL shipping.
4. Multiply the expected number of shipment lines from [Step 2](#) and the storage factor from [Step 3](#). _____
5. Enter the number of receipt lines you expect to be in the system during the time period specified in [Step 1](#). _____
6. Enter the receipt line multiplier. Choose one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system: _____
 - (a) 25 KB - for warehouses using no tag-controlled items, no serial tracking, and no LNPNs.
 - (b) 27 KB - for warehouses using no tag-controlled items, no serial tracking, and no LNPNs or only Pallet LPNs.
 - (c) 35 KB - for warehouses using no tag-controlled items, no serial tracking, and more than 80% CASE LPNs.
 - (d) 40 KB - for warehouses using tag-controlled items, serial tracking, and more than 80% CASE LPNs.
7. Multiply the expected number of receipt lines from [Step 5](#) and the storage factor from [Step 6](#). _____

Table 6–2 Steps for Disk Space Estimation for the Networked WMS Module - If you have both DOM and nWMS

8.	Enter the number of warehouses planned:	
	(a) Enter the number of stores or stock rooms planned.	_____
	(b) Enter the number of other warehouses planned.	_____
9.	Calculate the minimum space required for your set up based on the data in Step 8 and the minimum storage requirement given below:	
	(a) 20 MB for each store or stock room.	_____
	(b) 50 MB for each other warehouse.	_____
10.	The minimum operational storage requirements for the Sterling Multi-Channel Fulfillment Solution. For more information on the storage estimates refer to the " Note 3 ".	500 MB
11.	Enter the total estimated storage obtained by adding the values from Step 4 , Step 7 , Step 8 , Step 9 and Step 10 . For more information on the amount of usable space, refer to " Note 4 ".	_____
12.	Enter the value of Step 9 from Table 6–1 on page 71.	_____
13.	Enter the total estimated storage obtained by adding the values from Step 11 and Step 12 . For more information on the amount of usable space, refer to " Note 4 ".	_____

Table 6–3 Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS

1.	Enter the number of years worth of information to be kept in the system (retention time). For a more detailed example, refer to " Note 1 ".	_____
2.	Enter the number of shipment lines you expect to be in the system during the time period specified in Step 1 . For the different types of shipments, refer to " Note 2 ".	_____

Table 6–3 Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS

- | | |
|--|--------------|
| <p>3. Enter the shipment line multiplier. This factor includes demand-based replenishment. Choose from one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system:</p> <p>(a) 25 KB - for warehouses using no tag-controlled items, no serial tracking, no LNPNs or only Pallet LPNs, and more than 80% PARCEL shipping.</p> <p>(b) 27 KB - for warehouses using no tag-controlled items, no serial tracking, no LNPNs or only Pallet LPNs, and PARCEL as well as TL - LTL shipping.</p> <p>(c) 30 KB - for warehouses using tag-controlled items, no serial tracking, no LNPNs or only Pallet LPNs, and more than 80% TL - LTL shipping.</p> <p>(d) 35 KB - for warehouses using no tag-controlled items, no serial tracking, largely CASE LPNs, and more than 80% PARCEL shipping.</p> <p>(e) 40 KB - for warehouses using tag-controlled items, serial tracking, largely CASE LPNs, and PARCEL as well as TL - LTL shipping.</p> <p>(f) 50 KB - for warehouses using tag-controlled items, serial tracking, largely CASE LPNs, and more than 80% TL - LTL shipping.</p> | <p>_____</p> |
| <p>4. Multiply the expected number of shipment lines from Step 2 and the storage factor from Step 3.</p> | <p>_____</p> |
| <p>5. Enter the number of receipt lines you expect to be in the system during the time period specified in Step 1.</p> | <p>_____</p> |
| <p>6. Enter the receipt line multiplier. Choose from one of the following storage factors that most closely approximates a description of your Sterling Multi-Channel Fulfillment Solution system:</p> <p>(a) 25 KB - for warehouses using no tag-controlled items, no serial tracking, and no LNPNs.</p> <p>(b) 27 KB - for warehouses using no tag-controlled items, no serial tracking, and no LNPNs or only Pallet LPNs.</p> <p>(c) 35 KB - for warehouses using no tag-controlled items, no serial tracking, and more than 80% CASE LPNs.</p> <p>(d) 40 KB - for warehouses using tag-controlled items, serial tracking, and more than 80% CASE LPNs.</p> | <p>_____</p> |

Table 6–3 Steps for Disk Space Estimation for Networked WMS Module - If you have only nWMS

7.	Multiply the expected number of receipt lines from Step 5 and the storage factor from Step 6 .	_____
8.	Enter the number of warehouses planned:	
	(a) Enter the number of stores or stock rooms planned.	_____
	(b) Enter the number of other warehouses planned.	_____
9.	Calculate the minimum space required for your set up based on the data in Step 8 and the minimum storage requirement given below:	
	(a) 20 MB for each store or stock room.	_____
	(b) 50 MB for each other warehouse.	_____
10.	The minimum operational storage requirements for the Sterling Multi-Channel Fulfillment Solution. For more information on the storage estimates refer to " Note 3 ".	500 MB
11.	Enter the total estimated storage obtained by adding the values from Step 4 , Step 7 , Step 8 , Step 9 and Step 10 . For more information on the amount of usable space, refer to " Note 4 ".	_____

6.2.4 Tracking and Estimating Future Disk Requirements

You should track your actual database storage usage and the number of database records regularly. Correlating these two metrics enabled you to plan your future disk requirements. Moreover, determining the average amount of space used for each order line or shipment line, enables you to accurately predict your future growth requirements.

6.3 Installing Microsoft SQL Server 2000/2005 (Windows)

You can use a Microsoft SQL Server 2000/2005 database for maintaining information on the Sterling Multi-Channel Fulfillment Solution. When using a Microsoft SQL Server database with the Sterling Multi-Channel Fulfillment Solution, see [Chapter 2, "System Requirements"](#) for supported version information.

If you do not have SQL Server installed, follow the installation procedures in your SQL Server installation manual. Refer to the SQL Server documentation for information about creating the database,

including creating a schema repository, login, and tablespace. Be sure to install the correct version and patch.

Note: Ensure that Named Pipes & TCP/IP protocols are enabled in the network utility of the SQL Server.

Note: For SQL Server 2005, do not use case-sensitive column names in the database. Case-sensitive names will prevent the SQL Server 2005 System Management Console from loading.

6.3.1 Setting Database Parameters in SQL Server

For information about required parameter settings in your SQL Server database, see the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

6.3.2 SQL Server Database User Privileges

In SQL Server, you must grant DBO (Database Owner) permission to the Sterling Multi-Channel Fulfillment Solution user. The DB_DDLADMIN role is required for creating objects in the SQL Server database.

6.3.3 Configuring a SQL Server Database for a Production Environment

You need to configure your SQL Server database for running in a production environment with the Sterling Multi-Channel Fulfillment Solution. To configure a SQL Server database for a production environment, you must:

- [Size the database](#) by estimating the required disk space.
- Set the [database connection](#) properties.

6.3.3.1 Running Scripts for a SQL Server Database

To run the scripts:

1. Make sure you have a SQL Server client installed on your computer.

2. From the <INSTALL_DIR>\database\sqlserver\scripts directory, run the yfssqlserver_master_db_script.cmd script. This runs all of the required scripts using a sql server command-line utility.
3. Examine the log files for errors.

Note: If the application is installed on a non-Windows machine, all sql scripts under <INSTALL_DIR>\database\<dbtype>\scripts\CustomDBViews and yfs_seq_sqlserver.sql must be applied manually.

6.3.3.2 Enabling the Text Search Feature

To enable the text search feature on the SQL Server database:

1. Make sure that the Microsoft Search service is running on the machine on which the SQL Server is installed.

Note: By default, the full-text engine automatically runs as a service named Microsoft Search on Microsoft Windows, NT® Server 4.0, and Windows® 2000.

2. Log in to the SQL Server manager with a user ID having DBA privileges.

Note: The text search indexes that are created on the SQL Server using the <INSTALL_DIR>\repository\scripts\EFrame_TextIndexAdds.sql script are not automatically updated. Before running the EFrame_TextIndexAdds.sql script, the DBA must update the "*Database*" string in the EFrame_TextIndexAdds.sql script and specify the database name.

3. Verify that the text search index creation was successful.
4. From the <INSTALL_DIR>\repository\scripts directory, run the EFrame_TextIndexModify.sql script to enable the text search indexes to be incrementally updated when a text search enabled column is modified.

5. Edit the `customer_overrides.properties` file that is located in the `<INSTALL_DIR>\properties` directory to add the following entries:

```
yfs.yfs.db.textsearch=Y
```

For information on how to create the text search indexes, see the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.

6.3.4 Installing the JDBC Driver in SQL Server

The Sterling Multi-Channel Fulfillment Solution requires the correct Microsoft SQL Server driver. See [Chapter 2, "System Requirements"](#) for supported version information. The supported version of the JDBC driver builds the correct directory structure.

Go to <http://www.microsoft.com/downloads> to download the driver for your version of SQL Server, any appropriate patches, and the following .jar files. Be sure to place all the .jar files in the same directory:

- For MSSQL Server 2000:
 - msbase.jar
 - mssqlserver.jar
 - msutil.jar
- For MSSQL Server 2005:
 - sqljdbc.jar version 1.1 (do not download version 1.2; this is not supported)

The MSSQL Server 2000 files must be repacked as a single .jar file for the installer. Follow these steps to create one .jar file:

1. Identify all the vendor database jar files for the JDBC driver.
2. Create a temporary working directory (`mkdir wd; cd wd`).
3. Extract the contents of each file used for the JDBC driver using the `jar` utility into the temporary working directory (`jar xvf <jdbc.jar>` for each supplied jar file).

Note: Various Sterling Multi-Channel Fulfillment Solution scripts, such as the one used for loading the factory defaults, specify a `DB_DRIVER`. The `DB_DRIVER` specified must include *all* of these JAR files. The `DB_DRIVER` setting is located in `sandbox.cfg`. To make changes to the `DB_DRIVER` setting, edit and save the file, then run `setupfiles.sh`.

4. Bundle the files in the temporary working directory into one file using the `jar` utility (`jar cvf new.jar *`).
5. Record the absolute path to the `.jar` file you created on the Preinstallation Checklist.

6.4 Installing Oracle (Windows)

You can use an Oracle database for maintaining information on the Sterling Multi-Channel Fulfillment Solution. The following sections provide the necessary steps to install and configure an Oracle database for production.

To install Oracle:

Follow the steps below to install Oracle with single or multiple byte characters:

1. If you do not have Oracle installed, follow the installation procedures in your Oracle Installation manuals.
2. Run the create instance procedure. Use a character set appropriate for your desired language.

```
CHARACTER SET "UTF8"
```

3. Configure the `INIT<INSTANCE_NAME>.ORA` file for Oracle as follows:

```
open_cursors= <set to appropriate value>
```

For example, the minimum value for WebLogic equals number of threads (across all application servers) + (connection pool size X prepared statement pool size)

```
cursor_sharing=similar  
compatible=<10.2.0.3>
```

```
timed_statistics=true
db_block_size=8192
optimizer_mode=CHOOSE
```

If you are using multi-byte character set, set the following and restart Oracle:

```
nls_length_semantics=CHAR
```

Alternatively you can run:

```
alter session set nls_length_semantics = CHAR
```

prior to running any create table scripts.

Setting this attribute ensures that the field sizes are not impacted by the number of bytes a data type can store. For example, Varchar(40) would now be able to store 40 Japanese characters instead of 40/3 bytes in the UTF-8 character set.

Note: When you change the multi-byte character set to CHAR by setting `nls_length_semantics = CHAR`, Oracle reserves space equivalent to 'n' chars, which is more than 'n' bytes. Therefore, when you run the `dbverify.cmd` command, the reduced entries in table columns are printed in the `EFrame_Drops.lst` file.

4. Download the Oracle JDBC driver `ojdbc14.jar` from the Oracle website and copy to a well known location for reference during installation.

The Oracle JDBC driver can be found at:

http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/index.html

Download the `ojdbc14.jar` file for Oracle 10.2.0.3.

6.4.1 Oracle Database User Privileges

Unless specifically stated for a given task, the Sterling Multi-Channel Fulfillment Solution user does not require database administrator (DBA) privileges.

Following are some of the basic privileges that should be granted to the Sterling Multi-Channel Fulfillment Solution administrative user who is involved in creating and modifying the Oracle database:

- ALTER ANY SEQUENCE
- ALTER SESSION
- ALTER USER DEFAULT ROLE
- CREATE ANY SEQUENCE
- CREATE PROCEDURE
- CREATE SEQUENCE
- CREATE SESSION
- CREATE SYNONYM
- CREATE TABLE
- CREATE TRIGGER
- CREATE TYPE
- CREATE VIEW
- EXECUTE ANY PROCEDURE
- EXECUTE ANY TYPE
- GRANT "CONNECT"
- INSERT ANY TABLE
- SELECT ANY DICTIONARY
- SELECT ANY SEQUENCE
- SELECT ANY TABLE
- SELECT_CATALOG_ROLE
- UPDATE ANY TABLE

The following are some of the basic privileges that should be granted to the application user whose involvement is restricted just to running the application:

- CREATE SESSION
- ALTER SESSION

- EXECUTE ANY PROCEDURE
- INSERT ANY TABLE
- UPDATE ANY TABLE
- SELECT ANY TABLE
- SELECT ANY SEQUENCE

Note: If you are using text indexes, you must also have privileges for CTXAPP or CTXCAT, depending on the type of text indexes you are using.

6.4.2 Configuring an Oracle Database for Production (Windows)

You need to configure your Oracle database for running in a production environment with the Sterling Multi-Channel Fulfillment Solution. To configure an Oracle database for a production environment, you must:

- [Size the database](#) by estimating the required disk space.
- Create views and db_link or synonyms for integrating with the Sterling Warehouse Management System installation.
- Set the [database connection](#) properties.

To create the Oracle database to handle multiple byte characters:

1. Do not modify the Sterling Multi-Channel Fulfillment Solution DDL.
2. Choose the correct data encoding format for your language. See "[To install Oracle:](#)" on page 82 for more information.
3. Choose the character set suitable for your language. See "[To install Oracle:](#)" on page 82 for specific settings to ensure the database field sizes.

To set up scripts (if you are using locally managed tablespaces or another utility to size your database):

1. Create tablespaces where the Sterling Multi-Channel Fulfillment Solution tables and indexes reside.

2. Modify the
`<INSTALL_DIR>\repository\scripts\EFrame_TableChanges.sql`
file to reference your newly created tablespaces.

The DDLs in the Sterling Multi-Channel Fulfillment Solution scripts create a standard set of indexes. You may need to create additional indexes or modify existing indexes according to your business practice.

To run the scripts:

1. Log into the Oracle server manager as `sysdba`.
2. Create the user that is the designated schema owner.
3. Grant the privileges listed in [Section 6.4.1, "Oracle Database User Privileges"](#) on page 83 to the newly created user.
4. Log out of the Oracle Server Manager and log back in as the newly created user.
5. Verify the database as described in [Section 13.1.1.2, "Verifying the Database Schema"](#) on page 160.
6. Load the Sterling Multi-Channel Fulfillment Solution database factory defaults as described in [Section 13.1.1.3, "Loading the Sterling Multi-Channel Fulfillment Solution Database Factory Defaults After Installation"](#) on page 165.
7. Check for the degree of parallelism, using information from the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

6.4.2.1 Enabling the Text Search Feature

The Sterling Multi-Channel Fulfillment Solution supports two types of text search indexes on Oracle databases: CTXCAT and CONTEXT. The CTXCAT index supports automatic updating of text search indexes, whereas, the CONTEXT index does not support automatic updating of text search indexes. Sterling Commerce recommends that you use the CTXCAT index.

For information on how to create the text search indexes, refer to the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.

This section explains the following:

- [Enabling the Text Search Feature for CTXCAT Index](#)

- [Enabling the Text Search Feature for CONTEXT Index:](#)

6.4.2.1.1 Enabling the Text Search Feature for CTXCAT Index

The CTXCAT index automatically updates text search indexes. Therefore, the DBA need not manually run the `EFrame_TextIndexUpdates.sql` script to update text search indexes.

To enable the text search feature on an Oracle database using the CTXCAT index:

1. Make sure that the Oracle database is configured with the Oracle Text feature.
2. Log in to the Oracle server with a user ID having the CTXAPP privilege.
3. Verify that the text search index creation was successful.
4. Edit the `customer_overrides.properties` file that is located in the `<INSTALL_DIR>\properties` directory to add the following entries:

```
yfs.yfs.db.textsearch=Y
yfs.yfs.db.textsearch.oracle.contexttype=ctxcat
```

6.4.2.1.2 Enabling the Text Search Feature for CONTEXT Index:

The CONTEXT index does not automatically update text search indexes. Therefore, the DBA has to manually update text search indexes by running the `EFrame_TextIndexUpdates.sql` script.

To enable the text search feature on Oracle database using the CONTEXT index:

1. Make sure that the Oracle database is configured with the Oracle Text feature.
2. Log in to the Oracle server with a user ID having the CTXAPP privilege.

Note: The CONTEXT type text search indexes that are created on Oracle database using the <INSTALL_DIR>\repository\scripts\EFrame_TextIndexAdds.sql script are not updated automatically. The DBA has to run the EFrame_TextIndexUpdates.sql script to update the CONTEXT type text search indexes whenever required using scheduled jobs. The frequency of these scheduled jobs can be decided by the DBA.

3. Verify that the text search index creation was successful.
4. Edit the customer_overrides.properties file that is located in the <INSTALL_DIR>/properties/ directory to add the following entries:

```
yfs.yfs.db.textsearch=Y  
yfs.yfs.db.textsearch.oracle.contexttype=context
```

6.4.3 Using an Oracle Database Server (Windows)

You can use an Oracle 10g database with the Sterling Multi-Channel Fulfillment Solution. See [Chapter 2, "System Requirements"](#) for supported version information. The Sterling Multi-Channel Fulfillment Solution supports Oracle 10g in a single node database environment.

To use an Oracle 10g database:

- Create the database. Refer to the Oracle documentation for information about creating the database, including creating a schema repository, login, and tablespace. Be sure to install the correct version and patches.
- Configure the database by completing the following tasks:
 - Setting Database Parameters in Oracle
 - Rolling Back or Undoing Changes in Oracle
 - Granting Permissions in Oracle
 - Installing the JDBC Driver in Oracle

6.4.3.1 Setting Database Parameters in Oracle

For information about required parameter settings in your Oracle database, see the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

6.4.3.2 Rolling Back or Undoing Changes in Oracle

You can roll back or undo changes in Oracle using the following method:

(Oracle versions 10g or later) These versions support AUTO UNDO management. It is recommended that you use this option. This avoids manual monitoring of UNDO segments.

If a server is upgraded from Oracle 9i, set the UNDO_MANAGEMENT=AUTO parameter in init<SID>.ora. Your database administrator needs to determine the UNDO_RETENTION setting. Ensure that the file system which has the UNDOTBS1 tablespace has enough space to use the AUTOGROW setting.

6.4.4 Installing the JDBC Driver in Oracle (Windows)

The Sterling Multi-Channel Fulfillment Solution requires the appropriate JDBC driver for Oracle 10g databases. These drivers are thin client, 100% Pure Java™ JDBC drivers. See [Chapter 2, "System Requirements"](#) for supported version information.

The supported versions of the JDBC driver build the correct Sterling Multi-Channel Fulfillment Solution directory structure.

After you obtain the correct JDBC driver file, record the absolute path to its location on your system. You must supply this absolute path when you install the Sterling Multi-Channel Fulfillment Solution.

Installing in UNIX or Linux Environments

This chapter explains how to install the Sterling Multi-Channel Fulfillment Solution in UNIX or Linux environments. This chapter also provides the information required to complete [Step 7](#) of the “[Installation Checklist](#)” on page 1 for these environments.

You can install the Sterling Multi-Channel Fulfillment Solution on a UNIX or Linux system locally in an X Windows environment, or remotely, in a text-based console environment.

7.1 Before You Begin

If you are upgrading from a prior release, see the *Sterling Multi-Channel Fulfillment Solution Upgrade Guide* **before** continuing with the installation of the Sterling Multi-Channel Fulfillment Solution.

Before installing the Sterling Multi-Channel Fulfillment Solution, ensure that you have installed the appropriate software listed in [Chapter 2](#), “[System Requirements](#)”.

Throughout this document, `<INSTALL_DIR>` refers to the directory where you install the Sterling Multi-Channel Fulfillment Solution. For example, `<INSTALL_DIR>=/Supply_Chain_Apps`.

7.2 Preinstallation Setup for a UNIX or Linux Environment

The following topics will assist you with the preinstallation tasks when planning to install the Sterling Multi-Channel Fulfillment Solution in a UNIX or Linux environment:

- “[Key Terms \(UNIX/Linux\)](#)” on page 92

- ["Checklist for UNIX or Linux Preinstallation" on page 92](#)
- ["Checking System Requirements \(UNIX/Linux\)" on page 96](#)
- ["Installing the Java Software Development Kit \(UNIX/Linux\)" on page 96](#)
- ["Creating a UNIX Account" on page 96](#)
- ["Applying Database Definition Language \(DDL\) Statements \(UNIX/Linux\)" on page 96](#)
- ["About Silent Installations \(UNIX/Linux\)" on page 97](#)

7.2.1 Key Terms (UNIX/Linux)

The following terms and definitions will assist you in understanding the concepts discussed in this document:

- Database catalog name – This is typically the database name; also known as SERVICE_NAME or SID in some versions of Oracle.
- Java Software Development Kit (JDK) – Software development kit (SDK) for producing Java programs. Produced by Sun Microsystems, Inc., the JDK includes JavaBeans component architecture and support for JDBC.

7.2.2 Checklist for UNIX or Linux Preinstallation

The preinstallation checklist contains the items you need to gather, and tasks you need to complete prior to installing the Sterling Multi-Channel Fulfillment Solution.

Note: When creating a name, such as an account name, permissions name, profile name, or database name, follow these conventions:

- Use any valid alphanumeric characters and _ (underscore).
- Do not use spaces or apostrophes.

You may want to make a copy of the following checklist and use it to record the information you collect. The sections following the checklist included detailed explanations of some of the items on the checklist.

Preinstallation Checklist for UNIX or Linux

Step	Description	Your Notes
1	Verify that your system meets the hardware and software requirements specified for this release. For more information, refer to “Checking System Requirements (UNIX/Linux)” on page 96.	
2	If you are using a non-English environment, confirm that you are using the appropriate character set.	
3	Determine and record information about the JDK. <ul style="list-style-type: none"> • Version of the JDK • Absolute path to the JDK For more information, refer to “Installing the Java Software Development Kit (UNIX/Linux)” on page 96 and Chapter 2, “System Requirements”.	
4	Set Umask to 002.	
5	Determine and record information about your Oracle, MS SQL 2000/2005, or DB2 database server. Be aware that this information may be case sensitive. <ul style="list-style-type: none"> • Database user name and associated password • Database catalog name (For more information, see “Key Terms (UNIX/Linux)” on page 92.) • Database host name (or IP address) • Database host port number • Absolute path and file name for the JDBC driver • Version of the JDBC driver 	
6	Decide if you are going to manually or automatically apply database definition language (DDL) statements (schema) to the database. For more information, refer to “Applying Database Definition Language (DDL) Statements (UNIX/Linux)” on page 96.	

Preinstallation Checklist for UNIX or Linux

Step	Description	Your Notes
7	<p>Determine and record the directory in which you plan to install the Sterling Multi-Channel Fulfillment Solution.</p> <ul style="list-style-type: none">• The file system must have adequate free disk space.• The name of the directory is case-sensitive.• If you use the silent installation method, you cannot install into a pre-existing directory. The silent installation process will fail if a pre-existing directory is specified. See “About Silent Installations (UNIX/Linux)” on page 97 for more information about using the silent installation method.• If you use the GUI or text-based installation methods, you can install into either a pre-existing directory or a new directory to be created by the installation process.	

Preinstallation Checklist for UNIX or Linux

Step	Description	Your Notes
8	If you are running a silent installation, create your <code>silent_install.txt</code> property file. For more information, refer to “About Silent Installations (UNIX/Linux)” on page 97.	
9	<p>Note: This step is only for application systems that use the Linux operating system.</p> <p>Make the following system change:</p> <ol style="list-style-type: none"> 1. If the base locale for the system is English, set the <code>LANG</code> environment variable to <code>en_US</code>. 2. Reboot the system. 	
10	<p>Note: This step is only for application systems that use the RedHat Enterprise Linux operating system.</p> <p>Make the following system changes:</p> <ol style="list-style-type: none"> 1. If the base locale for the system is English, edit the <code>/etc/sysconfig/i18n</code> file by changing the <code>SUPPORTED</code> variable from <code>en_US.utf8</code> to <code>en_US</code>. You can also allow multiple support using the following format: <pre>en_US.utf8:en_US</pre> <p>Save and close the <code>/etc/sysconfig/i18n</code> file.</p> 2. Edit the <code>/etc/security/limits.conf</code> file by adding the following lines: <pre>* hard nofile 8196 * soft nofile 2048 * hard memlock 3000000 * soft memlock 4000000 * hard nproc 16000 * soft nproc 16000 * hard stack 512000 * soft stack 512000</pre> <p>This updates the system ulimits.</p> <p>Save and close the <code>/etc/security/limits.conf</code> file.</p> 3. Reboot the system. 	

7.2.3 Checking System Requirements (UNIX/Linux)

Before you begin, verify that your system meets the hardware and software requirements specified for Release 8.0 of the Sterling Multi-Channel Fulfillment Solution. The hardware requirements listed are the minimum required. For current information, see [Chapter 2, "System Requirements"](#).

7.2.4 Installing the Java Software Development Kit (UNIX/Linux)

You must install the Java Software Development Kit (JDK) and the patches specific to your system. You must supply the absolute path when installing the Java Software Development Kit (JDK). To determine which JDK version and patches you need, see [Chapter 2, "System Requirements"](#). After you install the JDK, record the absolute path to its location on your system.

7.2.5 Creating a UNIX Account

In a UNIX or Linux environment, you must create a UNIX administrative account on the host server for each installation of the Sterling Multi-Channel Fulfillment Solution. For example, if you want to create a test environment and a production environment, you need to create two UNIX accounts on the host server, one for the test and one for the production environment. For more information about creating UNIX accounts, see your operating system documentation.

7.2.6 Applying Database Definition Language (DDL) Statements (UNIX/Linux)

When you install the Sterling Multi-Channel Fulfillment Solution, you can manually apply database definition language (DDL) statements to your database tables instead of requiring the installation process to do it directly. This enables you to apply DDL statements for database creation separately from the installation.

This feature increases database security by reducing the database permissions of the Sterling Multi-Channel Fulfillment Solution database user. The rights to create tables, indexes, and so forth can be reserved for a secure user like a database administrator (DBA). A business can

require that only a DBA with the proper permissions can make database changes.

7.2.7 About Silent Installations (UNIX/Linux)

You can use a silent installation process, which automates part of the installation process and limits manual interaction with the installation program. This type of installation is detailed in [“Running the Installation Program in UNIX or Linux \(from Manually Edited Silent Install File\)”](#) on page 108.

7.2.7.1 Creating the Silent Installation File

To use the silent install process, you first create a silent installation file using a text editor. The file must contain the entries marked as required in the following table. When you are finished editing the file, record its name and location. You will use this information during the installation.

Example Entry	Description
INSTALL_DIR=<INSTALL_DIR>	(Required) Full path of your installation directory. Note: This property should not point to a pre-existing directory.
DB_VENDOR=<db_vendor>	(Required) The database vendor to use (Oracle, DB2, MSSQL).
MSSQL2005=<TRUE/FALSE>	If you are running on MSSQL 2005, this attribute must be set to TRUE. If you are running on MSSQL 2000, this attribute must be set to FALSE. If you are not running on MSSQL, do not include this attribute. NOTE: This attribute is case-sensitive.
DB_USER=<db_user_name>	(Required) User of database (system or user name).
DB_DATA=<db_dat_catalog>	(Required) Database name to connect with. (For more information, see “Key Terms (UNIX/Linux)” on page 92.)
DB_PASS=<db_password>	(Required) Database password.
DB_HOST=<db_host>	(Required) Host for database (for example, server or IP address).

Example Entry	Description
DB_PORT=<db_listener_port>	(Required) Database listener port.
DB_DRIVERS=<absolute path to driver jar>	(Required) Path to database driver file.
DB_DRIVERS_VERSION=<db_driver_version>	(Required) Free form version string for JDBC driver. This is informational only.
LOAD_FACTORY_SETUP=true	<p>Indicates whether you want to load factory setup defaults during installation (true) or manually after installation (false).</p> <p>For information about manually loading the factory defaults, see “Loading the Sterling Multi-Channel Fulfillment Solution Database Factory Defaults After Installation” on page 165.</p>
NO_DBVERIFY	<p>Valid values are true/false. When set to true during installation and installservice, dbverify will not be run. This means that the Sterling Multi-Channel Fulfillment Solution will not generate DDL to make the database like the XML entity repository.</p>
ACCEPT_LICENSE=Y	<p>Because the silent installer does not bring up an explicit license dialog, please specify your acceptance of the licensing terms by including this property in your silent install file.</p> <p>Note: To review the license file prior to installation, browse to the ProductFiles folder on your product CD and open Readme.htm.</p>
JAVADOC_PRODUCT_LABEL=Sterling Multi-Channel Fulfillment Solution 8.0	(Required) Specifies the Javadocs to be installed.

7.3 Installing Sterling Multi-Channel Fulfillment Solution in a UNIX or Linux Environment

Installing the Sterling Multi-Channel Fulfillment Solution in a UNIX or Linux environment includes the following sections:

- [“Running the Installation Program in UNIX or Linux \(GUI-Based\)”](#) on page 99
- [“Running the Installation Program in UNIX or Linux \(Text-Based\)”](#) on page 104
- [“Running the Installation Program in UNIX or Linux \(from Manually Edited Silent Install File\)”](#) on page 108

7.3.1 Before You Install: For AIX Installations Only

During the installation process, you specify the name of the directory to be created for the Sterling Multi-Channel Fulfillment Solution. The installation process creates the directory and uses it as the Home folder for the Sterling Multi-Channel Fulfillment Solution files and subdirectories. Throughout this book, this directory is referred to as `<INSTALL_DIR>`.

To ensure that `<INSTALL_DIR>` has the necessary permissions, AIX users must run the following command on the parent directory of `<INSTALL_DIR>` before installation:

```
chmod -R a-s <absolute path>/install_dir_parent
```

where `install_dir_parent` is the directory in which `<INSTALL_DIR>` will be created.

For example, to specify `AIX_1/applications/test1/my_install` as your installation directory, you could run the command from the `AIX_1/applications` directory (directly above the `test1` directory):

```
chmod -R a-s test1
```

or from another location on the file system:

```
chmod -R a-s /AIX_1/applications/test1
```

This ensures that when the `my_install` directory is created during installation, it inherits the correct permissions from `test1`.

7.3.2 Running the Installation Program in UNIX or Linux (GUI-Based)

Use the following instructions to install in a UNIX or Linux environment from a command line, using a graphical user interface (GUI) in an X Windows client.

Note: The following instructions assume that you received an installation CD. If you downloaded the Sterling Multi-Channel Fulfillment Solution or a Service Pack (SP) from the Electronic Software Distribution (ESD) Portal, unzip the downloaded file to an empty directory. The directory containing the unzipped files is an electronic image of an installation CD. Use this directory wherever there is a reference to the installation CD in the following instructions. Ignore any instructions to place the installation CD in a drive.

Note: During the installation, various messages are displayed, including some warning messages. These warning messages require no action on your part and are included so that helpful data is recorded in the log file.

To install the Sterling Multi-Channel Fulfillment Solution, refer to your preinstallation checklist and follow the steps below.

1. Place the Sterling Multi-Channel Fulfillment Solution installation CD in the appropriate drive.
2. From the installation CD, copy the `SCIInstallWizard.jar` and `SSCAP_8.0.jar` files to your home directory or base directory and change to that directory.

If you are using FTP to copy the files, verify that your session is set to binary mode.

3. Log in to a Windows machine.
 - a. Use a connectivity client to connect to your UNIX/Linux account.
 - b. Set the display to use your X server as a client using the following command:

```
export DISPLAY=<server>:0.0
```

(or the appropriate Display identifier)

Note: In the above command, :0.0 can be a different value, for example; :8.0.

4. If you are upgrading from a previous release, stop any running instances of the previous installation.
5. For either a new Sterling Multi-Channel Fulfillment Solution system, or for an upgrade from a previous release, enter the following command:

Note: On Linux, do not use any soft/symbolic links in the path to the jar file. Make sure that you specify the full path to the jar file.

Note: You **must** be in the directory where SCIInstallWizard.jar resides when issuing this command.

```
<JAVA_HOME>/bin/java -Xmx512m -jar SCIInstallWizard.jar
```

If you are doing an upgrade, at this point, refer to the *Sterling Multi-Channel Fulfillment Solution Upgrade Guide* to complete the installation.

The installation dialog box appears in a GUI.

6. Click Next to start the installation program.
7. Review the license agreement, and click Accept to accept the terms.
8. Type the full path of your JDK directory, or search for it using the Select Folder button. Click Next.
9. On the Upgrade/Database Options screen, do one of the following:
 - If you are upgrading from a previous release, or if you do not want the database DDLs and Factory Setup installation done as part of this installation process, check the *Do you wish to upgrade from a previous Install? Also check this option if you do NOT want Database DDLs and FactorySetup installed* option and click Next.

Note: If you check this option, you must manually create your database tables and load factory setup after the installation process. See the section about configuring your database for production in [Chapter 5, "Installing and Configuring Database Tier Software on UNIX/Linux"](#) for information about running view scripts after the initial installation and [Chapter 13, "Configuring Utilities"](#) for information about manually installing the database DDLs and factory setup.

- If you are **not** upgrading from a previous release, or if you **do** want the installation process to apply the database DDLs and install factory setup, leave the option unchecked and click Next.
10. Choose an installation directory for the Sterling Multi-Channel Fulfillment Solution. Click Select Folder and navigate to the folder you want to use as the installation directory. This directory is referred to as <INSTALL_DIR> in subsequent prompts. After selecting the folder, click Next.

If the directory does not exist, a message asking if the directory should be created is displayed. Click Yes to create the directory or No to return to the previous screen.

Note: The GUI installation creates the following installation directory structure:

<INSTALL_DIR>/Foundation

The installation process lays down the Sterling Multi-Channel Fulfillment Solution files and subdirectories under <INSTALL_DIR>/Foundation.

11. Select the database vendor that you want to use (Oracle, Microsoft SQL Server 2000, Microsoft SQL Server 2005, or DB2) and click Next.
12. Configure your database by entering the following information and click Next:
- Database user name
 - Database password
 - Confirm database password

- Database catalog name (For more information, see [“Key Terms \(UNIX/Linux\)”](#) on page 92.)
 - Database host name (or IP address)
 - Database port
 - JDBC driver file (Type or browse and select the absolute path and file name for the JDBC driver)
13. After you enter the database information and click Next, the Confirm Database Information screen displays the database account information you entered on the previous screen. This screen is read-only. If the information is correct, click Next. If any information needs to be changed, click Back to return to the previous screen and make the changes.
 14. The Pending Installation Tasks screen is displayed. This screen is read-only. The following tasks are shown:
 - *Verify Sufficient Disk Space*
 - *Install Foundation Components*
 - *Save install files*

The *Installation location* is also shown on the screen. This is the directory you chose earlier, followed by the subdirectory name "Foundation". The GUI installer installs the Sterling Multi-Channel Fulfillment Solution in the Foundation subdirectory, unlike the text-based and silent installations, which install the product directly into the folder you specify. Click Next.

15. On the Installation Progress screen, click Install to proceed with the installation. If you want to see detailed information about the progress of the installation, click Show Details, then click Install. This information will also be available after installation in the `<INSTALL_DIR>/PreInstallSI.log` file.
16. When the installation is finished, the message *Installation Wizard completed. Please see the installation guide for next steps* is displayed. Click OK to close the message box. The Installation Progress screen displays the status Complete as its heading and the message *BUILD SUCCESSFUL* in the Output box.

Note: Views must be created manually. Instructions to create views vary depending on what database is used. All database view related scripts are located at `<INSTALL_DIR>/database/<db_type>/scripts`. For more information, refer to the section about configuring your database type (Oracle or DB2) for production in [Chapter 5, "Installing and Configuring Database Tier Software on UNIX/Linux"](#).

7.3.3 Running the Installation Program in UNIX or Linux (Text-Based)

Use the following instructions to install in a UNIX or Linux environment from a command line, using a text-based (non-GUI) interface.

Note: The following instructions assume that you received an installation CD. If you downloaded the Sterling Multi-Channel Fulfillment Solution or a Service Pack (SP) from the Electronic Software Distribution (ESD) Portal, unzip the downloaded file to an empty directory. The directory containing the unzipped files is an electronic image of an installation CD. Use this directory wherever there is a reference to the installation CD in the following instructions. Ignore any instructions to place the installation CD in a drive.

Note: During the installation, various messages are displayed, including some warning messages. These warning messages require no action on your part and are included so that helpful data is recorded in the log file.

To install the Sterling Multi-Channel Fulfillment Solution, refer to your preinstallation checklist and follow the steps below.

1. Place the Sterling Multi-Channel Fulfillment Solution installation CD in the appropriate drive.

2. From the installation CD, copy the `SCIInstallWizard.jar` and `SSCAP_8.0.jar` files to your home directory or base directory and change to that directory.

If you are using FTP to copy the files, verify that your session is set to binary mode.
3. If you are upgrading from a previous release, stop any running instances of the previous installation.
4. To begin the installation, type the absolute path to the JDK directory followed by the command:

Note: On Linux, do not use any soft/symbolic links in the path to the jar file. Make sure that you specify the full path to the jar file.

Note: You **must** be in the directory where `SCIInstallWizard.jar` resides when issuing this command.

```
<JAVA_HOME>/bin/java -Xmx512m -jar SCIInstallWizard.jar
```

5. At the "Welcome to the Sterling Multi-Channel Fulfillment Solution Installer" prompt, press Enter.
6. At the "Press Enter to view the license agreement" prompt, press Enter to begin. The first page of the license agreement is displayed. Type N for the next page of the agreement, or scroll directly to the end of the agreement by pressing any other key. After the last page of the license agreement is displayed, the prompt "Do you accept the license? Y or N" is displayed. Press Y for yes, N for no.
7. At the "Select JDK directory" prompt, type the path of the JDK directory. This `<JDK_PATH>` should point to the root folder of the JDK installation. For example, the java executables should be present under the `<JDK_PATH>/bin` folder.
8. On the Upgrade/Database Options screen, you must respond to the following question, "Do you wish to upgrade from a previous Install? Also check this option if you do NOT want Database DDLs and FactorySetup installed." Use the following bullets as a guide for your answer:

- If you are upgrading from a previous release, type true and press Enter. See the Upgrade Guide for more specific information about performing an upgrade.
 - If you are NOT upgrading from a previous release, and you want the installation program to install the database DDLs and factory setup for you, type false and press Enter.
 - If you are NOT upgrading from a previous release, and you need to do the database DDL and factory setup installation manually after this installation program runs, type true and press Enter. You might need to choose this option if your DBA is the only one who can access or make changes to databases, but you are responsible for doing the rest of the installation, for example.
9. At the "Please choose an Installation directory" prompt, type the absolute path and name of the installation directory and press Enter. This can be either an existing or a new directory. If this is a new directory, you will be prompted, "The directory does not exist, create it?" Type Y for yes or N for no, then press Enter.

Note: The GUI installation creates the following installation directory structure:

<INSTALL_DIR>/Foundation

The installation process lays down the Sterling Multi-Channel Fulfillment Solution files and subdirectories under <INSTALL_DIR>/Foundation.

10. At the "Please select a database vendor" prompt, choose one of the following:

- (1) Oracle
- (2) MS SQL 2000
- (3) MS SQL 2005
- (4) DB2

Type the appropriate number and press Enter.

11. On the "Please enter your database account information" screen, you are prompted separately for each of the following items. Enter a

value for the first item, then press Enter. The second item will be displayed. Enter a value and press Enter. Repeat for each item.

- Database user name
- Database password
- Database catalog name (For more information, see [“Key Terms \(UNIX/Linux\)”](#) on page 92.)
- Database host name (or IP address)
- Database host port number (For DB2, use port 50000)
- JDBC driver file (Absolute path and file name for the JDBC driver - for DB2, use the Type-4 JDBC driver)

12. After you have entered the database information item and pressed Enter, the Confirm Database Information screen (read-only) displays the information you entered on the previous screen. Review each item's value, and press Enter to accept each one.

The installation program verifies the database connection. If a connection cannot be established, you receive an error and can re-enter the database information to make more connection attempts. If you still cannot make a connection, consult with your database administrator (DBA).

If the connection is successful, the installation process begins.

13. The system displays the message BUILD SUCCESSFUL when the installation is complete. For specific information about the installation, view the <INSTALL_DIR>/PreInstallSI.log file.

Note: Views must be created manually. Instructions to create views vary depending on what database is used. All database view related scripts are located at `<INSTALL_DIR>/database/<db_type>/scripts`. For more information, refer to the section about configuring your database type (Oracle or DB2) for production in [Chapter 5, "Installing and Configuring Database Tier Software on UNIX/Linux"](#).

7.3.4 Running the Installation Program in UNIX or Linux (from Manually Edited Silent Install File)

Use the following instructions to install in a UNIX or Linux environment from a command line, using a manually edited silent installation file.

Note: The following instructions assume that you received an installation CD. If you downloaded the Sterling Multi-Channel Fulfillment Solution or a Service Pack (SP) from the Electronic Software Distribution (ESD) Portal, unzip the downloaded file to an empty directory. The directory containing the unzipped files is an electronic image of an installation CD. Use this directory wherever there is a reference to the installation CD in the following instructions. Ignore any instructions to place the installation CD in a drive.

Note: During the installation, various messages are displayed, including some warning messages. These warning messages require no action on your part and are included so that helpful data is recorded in the log file.

To install the Sterling Multi-Channel Fulfillment Solution, refer to your preinstallation checklist and follow the steps below.

1. Place the Sterling Multi-Channel Fulfillment Solution installation CD in the appropriate drive.

2. From the installation CD, copy the `SSCAP_8.0.jar` to your home directory or base directory and change to that directory.
If you are using FTP to copy the file, verify that your session is set to binary mode.
3. Set up your silent installation file, using the guidelines in [“About Silent Installations \(UNIX/Linux\)”](#) on page 97. Record the path to your silent installation file.
4. Type one of the following commands, which include paths to the JDK, the application jar file, and the silent installation file:

Note: On Linux, do not use any soft or symbolic links in the path to the jar file. Make sure that you specify the full path to the jar file.

Note: The directory path to `SSCAP_8.0.jar` cannot include any spaces.

- If you are installing a new Sterling Multi-Channel Fulfillment Solution system, enter:

```
<JAVA_HOME>/bin/java -Xmx512m -jar SSCAP_8.0.jar -f
SilentInstallFile
```

The *SilentInstallationFile* is the one you created during preinstallation setup, as explained in [“About Silent Installations \(UNIX/Linux\)”](#) on page 97.

- If you are doing an upgrade, set `LOAD_FACTORY-SETUP=false` in your silent install file and enter:

```
<JAVA_HOME>/bin/java -Xmx512m -jar SSCAP_8.0.jar -f
SilentInstallFile -cluster
```

If the application is running, stop the previous installation before proceeding.

See the Upgrade Guide for more detailed information about performing a Sterling Multi-Channel Fulfillment Solution upgrade.

After the installation process begins, you can follow the progress of your installation through the

`<INSTALL_DIR>/PreInstallSI.log`.

The installation displays the message *Installation has completed successfully* when done.

Note: Views must be created manually. Instructions to create views vary depending on what database is used. All database view related scripts are located at `<INSTALL_DIR>/database/<db_type>/scripts`. For more information, refer to the section about configuring your database type (Oracle or DB2) for production in [Chapter 5, "Installing and Configuring Database Tier Software on UNIX/Linux"](#).

Installing in a Windows Environment

This chapter explains how to install the Sterling Multi-Channel Fulfillment Solution in a Windows environment. This chapter also provides the information required to complete [Step 7](#) of the “[Installation Checklist](#)” on page 1 for this environment.

8.1 Before You Begin

If you are upgrading from a prior release, see the *Sterling Multi-Channel Fulfillment Solution Upgrade Guide* **before** continuing with the installation of the Sterling Multi-Channel Fulfillment Solution.

Before installing the Sterling Multi-Channel Fulfillment Solution, ensure that you already have installed the appropriate software listed in [Chapter 2, “System Requirements”](#).

Throughout this document, <INSTALL_DIR> refers to the directory where you install the Sterling Multi-Channel Fulfillment Solution. For example, <INSTALL_DIR>=C:/Supply_Chain_Apps.

8.2 Preinstallation Setup for a Windows Environment

The following topics will assist you with the preinstallation tasks when planning to install the Sterling Multi-Channel Fulfillment Solution in a Windows environment:

- “[Key Terms \(Windows\)](#)” on page 112
- “[Checklist for Windows Preinstallation](#)” on page 112
- “[Checking System Requirements \(Windows\)](#)” on page 114

- ["Installing the Java Software Development Kit \(Windows\)"](#) on page 114
- ["About Silent Installations \(Windows\)"](#) on page 115

8.2.1 Key Terms (Windows)

The following terms and definitions will assist you in understanding the concepts of installing the Sterling Multi-Channel Fulfillment Solution in a Windows environment discussed in this document:

- Database catalog name – This is typically the database name; also known as SERVICE_NAME or SID in some versions of Oracle.
- Java Software Development Kit (JDK) – Software development kit (SDK) for producing Java programs. Produced by Sun Microsystems, Inc., the JDK includes JavaBeans component architecture and support for JDBC.

8.2.2 Checklist for Windows Preinstallation

The preinstallation checklist for the Windows environment identifies the prerequisite tasks you must complete before installing the Sterling Multi-Channel Fulfillment Solution.

Note: When creating a name, such as an account name, permissions name, profile name, or database name, follow these conventions:

- Use any valid alphanumeric characters and -, :, \$, &, or _.
- Do not use spaces or apostrophes.

You may want to make a copy of the following checklist and use it to record the information you collect for installing the Sterling Multi-Channel Fulfillment Solution:

Step	Description	Your Notes
1	Verify that your system meets the hardware and software requirements specified for Release 8.0. For more information, refer to "Checking System Requirements (Windows)" on page 114.	
2	If you are using a non-English environment, confirm that you are using the appropriate character set.	

Step	Description	Your Notes
3	<p>Determine and record information about the JDK.</p> <ul style="list-style-type: none"> • Version of the JDK • Absolute path to the JDK. The path name can not include spaces. <p>For more information, refer to “Installing the Java Software Development Kit (Windows)” on page 114.</p>	
4	<p>Determine and record information about your Oracle, SQL Server 2000/2005 or DB2 database server. Be aware that this information may be case sensitive.</p> <ul style="list-style-type: none"> • Database vendor • Database user name and associated password • Database catalog name (For more information, see “Key Terms (Windows)” on page 112.) • Database host name (or IP address) • Database host port number • Absolute path and file name for the JDBC driver • Version of the JDBC driver 	
5	<p>Decide if you are going to manually or automatically apply database definition language (DDL) statements (schema) to the database.</p> <p>For more information, refer to “Applying Database Definition Language (DDL) Statements (Windows)” on page 115.</p>	

Step	Description	Your Notes
6	<p>Determine and record the directory in which you plan to install the Sterling Multi-Channel Fulfillment Solution.</p> <ul style="list-style-type: none"> There must be a minimum of 5GB free disk space on the drive specified for installation The name of the directory can not include spaces and must be less than 30 characters long. If you use the silent installation method, you cannot install into a pre-existing directory. The silent installation process will fail if a pre-existing directory is specified. See “About Silent Installations (Windows)” on page 115 for more information about using the silent installation method. If you use the GUI installation method, you can install into either a pre-existing directory or a new directory to be created by the installation process. 	
7	<p>If you are running a silent installation, manually create your silent installation file. For more information, refer to “About Silent Installations (Windows)” on page 115.</p>	

8.2.3 Checking System Requirements (Windows)

Before you begin, verify that your system meets the hardware and software requirements specified for Release 8.0. The hardware requirements listed are the minimum required to run the Sterling Multi-Channel Fulfillment Solution. For current information, see [Chapter 2, “System Requirements”](#).

8.2.4 Installing the Java Software Development Kit (Windows)

You must install the Java Software Development Kit (JDK) and the patches specific to your system. To determine which JDK version and patches you need, see [Chapter 2, “System Requirements”](#). After you install the JDK, record the absolute path to its location on your system.

You must supply the absolute path when you install the Sterling Multi-Channel Fulfillment Solution.

Caution: In Windows, the directory name where the JDK resides cannot include a space.

The JavaHome key values need to be changed to equal the directory where you installed JDK.

8.2.5 Applying Database Definition Language (DDL) Statements (Windows)

When you install the Sterling Multi-Channel Fulfillment Solution, you can manually apply database definition language (DDL) statements to your database tables instead of requiring the installation process to do it directly. This enables you to apply DDL statements for database creation separately from the installation.

This feature increases database security by reducing the database permissions of the Sterling Multi-Channel Fulfillment Solution database user. The rights to create tables, indexes, etc. can be reserved for a secure user like a database administrator (DBA). A business can require that only a DBA with the proper permissions can make database changes.

If you choose to manually apply the DDL, the installation process will provide the location of the DDL scripts. The installation process will then continue the rest of the installation. The installation process may validate the database with DBVerify and warn you if there are differences, but it will not exit. It will allow the processing of the packages to continue normally.

If you do not choose to manually apply the DDL, the installation will apply both the DDL and the resources.

8.2.6 About Silent Installations (Windows)

You can create and use a silent installation process, which automates part of the installation process and limits manual interaction with the installation program. This type of installation is detailed in ["Running the](#)

[Installation Program in Windows \(Manually Edited Silent Install File\)](#)" on page 122.

8.2.6.1 Creating the Silent Installation File

You create the silent installation file using a text editor. The file must contain the entries marked as required in the following table. When you are finished editing the file, record its name and location. You will use this information during the installation.

Note: Use UNIX slashes ("/") when specifying paths in the file. For example, to specify the installation directory to be created, you might enter something similar to one of the following sample paths:

C:/Sterling80

or

D:/Applications/Myinstall

Example Entry	Description
INSTALL_DIR=<INSTALL_DIR>	(Required) Directory in which to install. Note: This property cannot point to a pre-existing directory, or the installation will fail.
DB_VENDOR=<db_vendor>	(Required) The database vendor to use (Oracle, DB2, MSSQL).
MSSQL2005=<TRUE/FALSE>	If you are running on MSSQL 2005, this attribute must be set to TRUE. If you are running on MSSQL 2000, this attribute must be set to FALSE. If you are not running on MSSQL, do not include this attribute. NOTE: This attribute is case-sensitive.
DB_USER=<db_user_name>	(Required) Database login ID with which to connect.

Example Entry	Description
DB_DATA=<db_dat_catalog>	(Required) Database name to connect with. (For more information, see “Key Terms (Windows)” on page 112.)
DB_PASS=<db_password>	(Required) Database password with which to connect.
DB_HOST=<db_host>	(Required) Host for database (for example, server or IP address).
DB_PORT=<db_listener_port>	(Required) Database listener port to which to connect.
DB_DRIVERS=<absolute path to driver jar>	(Required) Full path to JDBC driver file.
DB_DRIVERS_VERSION=<db_driver_version>	(Required) Free form version string for JDBC driver. This is informational only.
LOAD_FACTORY_SETUP=true	<p>Indicates whether you want to load factory setup defaults during installation (true) or manually after installation (false).</p> <p>For information about manually loading the factory defaults, see “Loading the Sterling Multi-Channel Fulfillment Solution Database Factory Defaults After Installation” on page 165.</p>
NO_DBVERIFY	<p>Valid values are true/false. When set to true during installation and installservice, dbverify will not be run. This means that the Sterling Multi-Channel Fulfillment Solution will not generate DDL to make the database like the XML entity repository.</p>
ACCEPT_LICENSE=Y	<p>Because the silent installer does not bring up an explicit license dialog, please specify your acceptance of the licensing terms by including this property in your silent install file.</p> <p>Note: To review the license file prior to installation, browse to the ProductFiles folder on your product CD and open Readme.htm.</p>
JAVADOC_PRODUCT_LABEL= Sterling Multi-Channel Fulfillment Solution 8.0	(Required) Specifies the Javadocs to be installed.

8.3 Installing Sterling Multi-Channel Fulfillment Solution in a Windows Environment

Installing the Sterling Multi-Channel Fulfillment Solution in a Windows environment includes the following sections:

- ["Running the Installation Program in Windows \(GUI-Based\)" on page 119](#)
- ["Running the Installation Program in Windows \(Manually Edited Silent Install File\)" on page 122](#)

8.3.1 Running the Installation Program in Windows (GUI-Based)

Note: The following instructions assume that you received an installation CD for the Sterling Multi-Channel Fulfillment Solution. If you downloaded the Sterling Multi-Channel Fulfillment Solution or a Service Pack (SP) from the Electronic Software Distribution (ESD) Portal, unzip the downloaded file to an empty directory. Do not change the directory structure of the newly unzipped files. The directory containing the unzipped files is an electronic image of an installation CD. Use this directory wherever there is a reference to the installation CD in the following instructions. Ignore any instructions to place the installation CD in a drive.

Note: To install more than one instance of the Sterling Multi-Channel Fulfillment Solution on the same Windows server, you must install the second instance in a different directory and use a different initial port number. This second port number must be at least 100 higher than the first port number.

During the installation, various messages are displayed, including some warning messages. These warning messages require no action on your part and are included so that helpful data is recorded in the log file.

To install the Sterling Multi-Channel Fulfillment Solution in a Windows environment, refer to your preinstallation checklist and follow the steps below.

1. Close all open Windows programs and any command prompt windows.
2. Copy the `SCIInstallWizard.jar` and `SSCAP_8.0.jar` files from your installation CD to a folder on your hard drive.
3. Start the installation process. From a command prompt, enter the following command:

Note: You **must** be in the directory where SCIInstallWizard.jar resides when issuing this command.

```
<JAVA_HOME>\bin\java.exe -Xmx512m -jar SCIInstallWizard.jar
```

The installation dialog box appears.

4. Click Next to start the installation program.
5. Review the license agreement, and click Accept to accept the terms.
6. Type the path of your JDK directory, or search for it using the Select Folder button. Click Next.
7. On the Upgrade/Database Options screen, do one of the following:
 - If you are upgrading from a previous release, or if you do not want the database DDLs and Factory Setup installation done as part of this installation process, check the *Do you wish to upgrade from a previous Install? Also check this option if you do NOT want Database DDLs and FactorySetup installed* option and click Next.

Note: If you check this option, you must manually create your database tables and load factory setup after the installation process. See the section about configuring your database for production in [Chapter 6, "Installing and Configuring Database Tier Software on Windows"](#) for information about running view scripts after the initial installation and [Chapter 13, "Configuring Utilities"](#) for information about manually installing the database DDLs and factory setup.

- If you are **not** upgrading from a previous release, or if you do want the installation process to apply the database DDLs and install factory setup, leave the option unchecked and click Next.
8. Choose an installation directory for the Sterling Multi-Channel Fulfillment Solution. Click Select Folder and navigate to the folder you want to use as the installation directory. For Windows 2000/NT server, the path to this directory must be 8 characters or less and cannot include any spaces. This directory is referred to as <INSTALL_DIR> in subsequent prompts. After selecting the folder, click Next.

If the directory does not exist, a message asking if the directory should be created is displayed. Click Yes to create the directory or No to return to the previous screen.

Note: The GUI installation creates the following installation directory structure:

<INSTALL_DIR>\Foundation

The installation process lays down the Sterling Multi-Channel Fulfillment Solution files and subdirectories under <INSTALL_DIR>\Foundation.

9. Select the database vendor that you want to use (Oracle, Microsoft SQL Server 2000, Microsoft SQL Server 2005, or DB2) and click Next.
10. Configure your database by entering the following information and click Next:
 - Database user name
 - Database password
 - Confirm database password
 - Database catalog name (For more information, see ["Key Terms \(Windows\)" on page 112.](#))
 - Database host name (or IP address)
 - Database port
 - JDBC driver file (Type or browse and select the absolute path and file name for the JDBC driver)
11. After you enter the database information and click Next, the Confirm Database Information screen displays the database account information you entered on the previous screen. This screen is read-only. If the information is correct, click Next. If any information needs to be changed, click Back to return to the previous screen and make the changes.
12. The Pending Installation Tasks screen is displayed. This screen is read-only. The following tasks are shown:
 - *Verify Sufficient Disk Space*

- *Install Foundation Components*
- *Save install files*

The *Installation location* is also shown on the screen. This is the directory you chose earlier, followed by the subdirectory name "Foundation". The Windows GUI installer installs the Sterling Multi-Channel Fulfillment Solution in the Foundation subdirectory, unlike the silent installation, which installs the product directly into the folder you specify. Click Next.

13. On the Installation Progress screen, click Install to proceed with the installation. If you want to see detailed information about the progress of the installation, click Show Details, then click Install. This information will also be available after installation in the <INSTALL_DIR>\PreInstallSI.log file.
14. When the installation is finished, the message *Installation Wizard completed. Please see the installation guide for next steps* is displayed. Click OK to close the message box. The Installation Progress screen displays the status Complete as its heading and the message *BUILD SUCCESSFUL* in the Output box.

Note: Views must be created manually. Instructions to create views vary depending on what database is used. All database view related scripts are located at <INSTALL_DIR>\database\<db_type>\scripts. For more information, refer to the section about configuring your database type (MSSQL 2000/2005, Oracle 10g, or DB2) for production in [Chapter 6, "Installing and Configuring Database Tier Software on Windows"](#).

8.3.2 Running the Installation Program in Windows (Manually Edited Silent Install File)

Use the following instructions to install in a Windows environment from a command line, using a manually edited silent installation file.

Note: The following instructions assume that you received an installation CD. If you downloaded the Sterling Multi-Channel Fulfillment Solution or a Service Pack (SP) from the Electronic Software Distribution (ESD) Portal, unzip the downloaded file to an empty directory. The directory containing the unzipped files is an electronic image of an installation CD. Use this directory wherever there is a reference to the installation CD in the following instructions. Ignore any instructions to place the installation CD in a drive.

Note: During the installation, various messages are displayed, including some warning messages. These warning messages require no action on your part and are included so that helpful data is recorded in the log file.

To install the Sterling Multi-Channel Fulfillment Solution, refer to your preinstallation checklist and follow the steps below:

1. Place the Sterling Multi-Channel Fulfillment Solution installation CD in the appropriate drive.
2. From the installation CD, copy the `SSCAP_8.0.jar` to your home directory or base directory and change to that directory.

If you are using FTP to copy the file, verify that your session is set to binary mode.
3. Set up your silent installation file, using the guidelines in [“About Silent Installations \(Windows\)”](#) on page 115. Record the path to your silent installation file.
4. At a command prompt, type one of the following commands, which include paths to the JDK, the `SSCAP_8.0.jar` file, and the silent installation file:

Note: The directory path to SSCAP_8.0.jar cannot include any spaces.

- If you are installing a new Sterling Multi-Channel Fulfillment Solution system, enter the following:

```
<JAVA_HOME>\bin\java -Xmx512m -jar SSCAP_8.0.jar -f SilentInstallFile
```

The Silent Installation File is the one you created during preinstallation setup, as explained in [“Creating the Silent Installation File”](#) on page 116.

- If you are doing an upgrade set LOAD_FACTORY_SETUP=false in your silent install file and enter the following:

```
<JAVA_HOME>\bin\java -Xmx512m -jar SSCAP_8.0.jar -f SilentInstallFile  
-cluster
```

After the installation process begins, you can follow the progress of your installation through the <INSTALL_DIR>\InstallSI.log file.

The installation displays the message *Installation has completed successfully* when done.

Note: Views must be created manually. Instructions to create views vary depending on what database is used. All database view related scripts are located at <INSTALL_DIR>\database\<db_type>\scripts. For more information, refer to the section about configuring your database type (MSSQL 2000/2005, Oracle 10g, or DB2) for production in [Chapter 6, “Installing and Configuring Database Tier Software on Windows”](#).

Installing the Print Server

This chapter explains how to install and configure the Software Label Manager (LLM) and Software Print Server (LPS).

This chapter also provides the information required to complete [Step 8](#) indicated on the [Installation Checklist](#) on page 1.

For more information about configuring the Software Label Manager and Print Server, see the *Software Label Manager User's Guide* and the *Software Print Server User's Guide*.

For more information about Performance Considerations for setting up the Software Print Server (LPS) see the *Software Print Server User's Guide*.

9.1 Installation of Software Components

The Software Print Server manages bar code label print requests between applications and hundreds of networked printers. As a general guideline, you should configure a maximum of 200 printers for each Software Print Server you install. For more information about server requirements and installation instructions, see the *Software Print Server User's Guide*. Contact your Software support representative for additional sizing and configuration support.

The Software Label Manager, used for designing labels, may be installed on any compatible PC. For more information about server requirements and installation guidelines, see the *Software Label Manager User's Guide*.

The Sterling Multi-Channel Fulfillment Solution supports printing in the following modes:

- File Copy Mode

- TCP/IP Sockets Mode

The `yfs.loftware.tcpip.sockets` attribute in the `yfs.properties.in` file determines the mode used for printing. By default this boolean property is set to 'N' for File Copy Mode.

To configure the Software printing in the TCP/IP Sockets Mode, edit the `customer_overrides.properties` file to include the following entry:

```
yfs.yfs.loftware.tcpip.sockets=Y
```

The Sterling Multi-Channel Fulfillment Solution requires the following settings in the Software Print Server Configuration Utility:

- In Directory Set up, ensure that the 'Pass Files' option is selected.
- **When using File Copy Mode:** In Directory Set up, ensure that the 'Enable Polling (Disable Event File Trigger)' option is selected. It is recommended that the Poll Interval value is set to 500 Milliseconds.

Note: In File Copy Mode, SAMBA should be configured when using a UNIX version of the application server.

The Drop Directories of the printers configured in Software need to be mounted on to the UNIX server using SAMBA.

9.2 Define Printers on Software

Configure printers on Software using the Software Design 32 tool. For more information about configuring printers using the Software Design 32 tool, see the *Software Label Manager User's Guide*.

9.3 Define Printers for the Sterling WMS Installation

For more information about configuring printers for the Sterling WMS, see the *Sterling Warehouse Management System Configuration Guide*.

9.4 Copying the Sterling WMS Standard Label Formats

The Sterling WMS provides Software Label Manager template (*.lwl) files which should be copied in the directory set up for labels using the Software Design 32 tool.

Also, copy the YCP_LABEL_FIELDS.LST file to the directory where the Software Label Manager has been installed. This file is available in the <INSTALL_DIR>/repository/xapi/template/source/<application_install>/prints/label directory.

9.5 Installation of JasperReports

JasperReports is an open source Java reporting tool that delivers rich content on the screen, to the printer or in the format of a PDF, HTML etc,. You can use JasperReports with the Sterling Multi-Channel Fulfillment Solution for printing or generating PDF objects for order reports, labels and so forth. The installation procedure and sample files are located in <INSTALL_DIR>/xapidocs/code_examples/jasperreports directory.

Note: For JasperReports, the Sterling Multi-Channel Fulfillment Solution uses the jasperreports-1.2.0.jar file. For more information about JasperReports and supporting jars and components, see the `readme.html` file located either in the <INSTALL_DIR>/xapidocs/code_examples/jasperreports directory.

Installing the Weighing Scale

Weighing scales are typically used at packing or manifest stations. This chapter describes the installation of weighing scales for use with the Sterling WMS.

This chapter also provides the information required to complete [Step 9](#) indicated on the ["Installation Checklist"](#) on page 1.

For more information regarding the Mettler-Toledo PS Weighing Scale, see the *Mettler-Toledo PS Weighing Scale User Guide*.

10.1 Installation of the Weighing Scale

The weighing scale is installed at each pack or manifest station requiring weighing scale integration.

To install the weighing scale, follow these steps on each client machine:

1. Launch your Internet Explorer browser.
2. In the Address bar, type
`http://<hostname>:<portnumber>/yantra/yfscommon/win32com.dll`
and press Enter. The File Download window appears.
3. Choose Save. The Save As window appears.
4. Save the file in any directory present in the System Class path. For example, on Windows NT, go to `C:/WINNT/system32`.

For more information about setting up the weighing scale and associating it with a station, see the Equipment section of the *Sterling Warehouse Management System Configuration Guide*.

For more information about system requirements, see [Chapter 2, "System Requirements"](#).

Installing the Sterling Multi-Channel Fulfillment Solution Mobile Application

This chapter describes how to install the Sterling Multi-Channel Fulfillment Solution Mobile Application for use on PocketPC, WinCE, and VT220 mobile terminals.

This chapter also provides the information required to complete [Step 10](#) indicated on the "[Installation Checklist](#)" on page 1.

For information on PocketPC and WinCE mobile terminal system requirements, see [Chapter 2, "System Requirements"](#).

Important: Install the Microsoft .NET Compact Framework on your local PC, BEFORE installing the Sterling Multi-Channel Fulfillment Solution Mobile Application. For more information about the supported versions for the Microsoft .NET Compact Framework, see [Section 2.7, "Sterling Multi-Channel Fulfillment Solution Mobile Application Requirements"](#). This may be downloaded from <http://www.microsoft.com>.

Installing the Microsoft .NET Compact Framework creates multiple .CAB files on the system, for multiple operating systems and processors of the device.

11.1 Installation on Mobile Terminals

To set up the PC with the Sterling Multi-Channel Fulfillment Solution Mobile Application for the Symbol Mobile Terminal, follow these steps:

1. Connect the Mobile Terminal to the PC that has ActiveSync installed. For more information, refer to the Communications section of the *Symbol Installation Guide*.

Note: It is suggested that ActiveSync be used to copy the files.

Alternatively, you may transfer the file over the network, if the device is already configured to access the LAN, or you may serve the .CAB file through a webserver and use the Internet Explorer browser on the device to download it.

If the Sterling WMS is installed on a UNIX server, it may be required to copy the files from the UNIX server to the PC before launching ActiveSync.

2. Choose Start > Programs > ActiveSync on the PC.
3. Choose File > Get Connected, if not already connected.
4. Choose the Explore icon. This brings up the File Explorer for the Mobile Terminal.

5. Go to the <INSTALL_DIR> folder.
6. In the File Explorer window, click the Folders icon to bring up folders in the left panel.
7. Copy the YantraMobileApp_xxx.CAB files from the <INSTALL_DIR>/mobileapp folder on the PC to the \Application folder on the mobile terminal.

Here, xxx refers to PPC.ARM, PPC.ARMV4, WCE4.ARMV4, or WCE4.ARMV4T.

Ensure that you choose the .CAB file that is relevant to the operating system and processor of your mobile terminal.

8. Double-click on the YantraMobileApp_xxx.CAB file, on your mobile terminal. This installs the application on the mobile terminal.

Note: This file is automatically deleted upon successful installation.

9. The sample file <INSTALL_DIR>/mobileapp/yantrahostlist.xml includes application servers with Loopback, Production, QA and Test names. Replace these with the application servers along with their IP addresses and port numbers. The application server names entered here are listed in the Servers drop-down list to which you can get connected when you launch the Sterling Multi-Channel Fulfillment Solution Mobile Application on the PocketPC or WinCE mobile terminals.

If you want to run the application on https, configure securemode and provide secureURL attributes in the <INSTALL_DIR>/mobileapp/yantrahostlist.xml file. The applicable values of securemode are 'all', 'loginonly', and 'none'.

If you set securemode to:

- 'all', the application runs on https.
- 'loginonly', only the login page runs on https and the rest of the application on http.

- 'none', secureURL is ignored and the application runs on http.

Note: To use the above listed attributes, refer to the `YantraHostList.xml` file provided with the application.

Note: This step is not valid for VT220 mobile terminal.

Note: Servers include application servers used for production, test, and other environments, if applicable.

10. Locate the properties file for your mobile device as specified in [Table 11–1](#) and rename that file to `YMAProperties.xml`.

Table 11–1 Device and Properties File

Device	Properties File
LXE MX7 handheld	<code>YMAProperties.MX7.xml</code>
LXE VX3X series truck mount	<code>YMAProperties.VX3X.xml</code>
PocketPC mobile terminal	<code>YMAProperties.ppc.xml</code>
Symbol VRC7900 series truck mount	<code>YMAProperties.vrc7900.xml</code>

11. Stop ActiveSync.

For additional information about the replication of the Sterling Multi-Channel Fulfillment Solution Mobile Application to multiple mobile devices, see the *Symbol Device Installation* documentation.

The *Symbol Device Installation* documentation also provides additional details on re-installation.

Note: To display clear and appropriate error messages, a locale specific "System_SR_<locale>.cab" resource file is needed. For more information about the locale specific resource file, contact your PocketPC product support at <http://support.microsoft.com>.

Note: Sterling Commerce recommends you to configure the barcode scanner such that the TAB character is suffixed with the scanned barcode data.

11.1.1 Ensuring Re-Installation on Cold Boot

This section provides instruction on how to automatically install Microsoft .NET Compact Framework and Sterling Multi-Channel Fulfillment Solution Mobile Application on cold-boot on various mobile terminals.

11.1.1.1 On a PocketPC Mobile Terminal

To ensure that the Microsoft .NET Compact Framework and the Sterling Multi-Channel Fulfillment Solution Mobile Application are installed automatically on cold boot, the following instructions must be followed as a one-time measure:

1. Install the Microsoft .NET Compact Framework on the mobile terminal, BEFORE installing the Sterling Multi-Channel Fulfillment Solution Mobile Application. This may be downloaded from <http://www.microsoft.com>.
2. Install the Sterling Multi-Channel Fulfillment Solution Mobile Application. For more information, see [Section 11.1, "Installation on Mobile Terminals"](#) on page 132.
3. Copy the Microsoft .NET Compact Framework installation CAB file from the local PC to the \Application folder on the mobile device.
4. Copy all the files under <INSTALL_DIR>/mobileapp to the \Application\YantraMobileApp folder on the mobile device.
5. Copy the file YantraMobileApp.lnk from the \Windows\Start Menu\Programs folder to the <INSTALL_DIR> folder.

6. Edit the following line in the `yantra.cpy` file located in the `<INSTALL_DIR>/mobileapp` folder:

```
\Application\netcf.core.ppc3.ARM.cab >  
\Windows\startup\netcf.core.ppc3.ARM.cab
```

Here, rename the `netcf.core.ppc3.ARM.cab` file to the CAB file name as appropriate for your handheld device.

7. Change all occurrences of Program Files to `<INSTALL_DIR>`, as applicable in all lines of the `yantra.cpy` file.
8. Save the modified `yantra.cpy` file under the `\Application` folder on the mobile device.

This ensures that the Microsoft .NET Compact Framework and the Sterling Multi-Channel Fulfillment Solution Mobile Application are installed on cold boot. When cold booting the system, the `yantra.cpy` file copies the installation files to the start directory.

11.1.1.2 On a Symbol VRC7900 WinCE Mobile Terminal

Symbol VRC7900 supports installation of software during cold boot by storing the installation .cab files in the `\FlashFx\CAB` folder.

All files that need to be copied to the folder `<INSTALL_DIR>/mobileapp` folder in the RAM file system must be placed under `\FlashFx\CopyToRam\Root\<INSTALL_DIR>/mobileapp`.

To ensure that the Microsoft .NET Compact Framework and the Sterling Multi-Channel Fulfillment Solution Mobile Application are installed automatically on cold boot, the following instructions must be followed as a one-time measure:

1. Install the Microsoft .NET Compact Framework on the mobile terminal, BEFORE installing the Sterling Multi-Channel Fulfillment Solution Mobile Application. This may be downloaded from <http://www.microsoft.com>.
2. Install the Sterling Multi-Channel Fulfillment Solution Mobile Application. For more information, see [Section 11.1, "Installation on Mobile Terminals"](#) on page 132.
3. Copy the Microsoft .NET Compact Framework installation CAB file, `netcf.core.WINCE.ARMV4.cab`, from the local PC to the `\FlashFx\CAB` folder on the mobile device.

4. Copy all the files (including the hidden file vsd_setup.dll) under the <INSTALL_DIR>\mobileapp folder to the \FlashFx\CopyToRam\Root\<INSTALL_DIR>\mobileapp folder.
5. If a desktop shortcut has been created, copy the corresponding shortcut file (with .lnk extension) to the \FlashFx\CopyToRam\System\Desktop folder.
6. Copy the \Windows\My Company YantraMobileApp.unload file to the \FlashFx\CopyToRam\System folder.

This ensures that the Microsoft .NET Compact Framework and the Sterling Multi-Channel Fulfillment Solution Mobile Application are installed on cold boot.

11.1.1.3 On a Denso BHT400B Win CE 5.0 Mobile Terminal

To ensure that the Microsoft .NET Compact Framework and the Sterling Multi-Channel Fulfillment Solution Mobile Application are automatically installed on cold-boot, follow these instructions as a one-time measure:

1. Install the Microsoft .NET Compact Framework 2.0 SP1 on the mobile terminal, prior to installing the Sterling Multi-Channel Fulfillment Solution Mobile Application. This can be downloaded from <http://www.microsoft.com>.
2. Install the Sterling Multi-Channel Fulfillment Solution Mobile Application. For more information about installing the Sterling Multi-Channel Fulfillment Solution Mobile Application, see [Section 11.1, "Installation on Mobile Terminals"](#).

Note: Cold-booting of the mobile terminal erases the data stored in RAM. Therefore, copy the installation files into a folder whose contents are retained even after performing a cold-boot. See the Mobile Terminal Operator's guide provided by the manufacturer to identify the appropriate folder.

For Win CE 5.0 mobile terminals manufactured by Denso Corporation, contents of the 'Flash' folder are retained after cold-booting. If you are using any other mobile terminal, locate the folder and replace all occurrences of 'Flash' with the located folder in the following steps:

3. Copy the Microsoft .NET Compact Framework 2.0 SP1 installation CAB file from the local PC to the \Flash folder on the mobile terminal.
4. Copy the relevant Sterling Multi-Channel Fulfillment Solution Mobile Application CAB file from the local PC to the \Flash folder on the mobile terminal.
5. Create the \MobileApp folder under \Flash folder. Copy the following files from the directory where the Sterling Multi-Channel Fulfillment Solution Mobile Application is installed (\Program Files\MobileApp) to \Flash\MobileApp folder:
 - YantraHostList.xml
 - YMAProperties.xml
6. Edit the yantra_wce50.bat file located in the <INSTALL_DIR>/mobileapp folder for the following changes:
 - Change all the occurrences of NETCFv2.wce4.ARMV4.cab to the Microsoft .NET Compact Framework 2.0 SP1 CAB file appropriate for your mobile terminal.
 - Change all occurrences of YantraMobileApp_WCE4.ARMV4.CAB to the Sterling Multi-Channel Fulfillment Solution Mobile Application CAB file that is appropriate for your mobile terminal.
7. Save the modified yantra_wce50.bat file under the \Flash\StartUp folder on the mobile terminal. The contents of the StartUp folder are run automatically when you perform a cold boot. Refer to the Mobile Terminal Operator's guide provided by the manufacturer to identify this folder.

This ensures that the Microsoft .NET Compact Framework and the Sterling Multi-Channel Fulfillment Solution Mobile Application are installed on cold-booting the mobile terminal.

11.2 Installing on VT220 Mobile Terminals

This section describes how to install the Sterling Multi-Channel Fulfillment Solution Mobile Application for use on VT220 mobile terminals.

The Sterling Multi-Channel Fulfillment Solution Mobile Application can be accessed from any VT220 emulation terminal.

Sterling Multi-Channel Fulfillment Solution Mobile Application requires the installation of `ncurses` to enable you to change the function key sequence mapping.

11.2.1 Installing `ncurses`

The VT220 client requires `infocmp` and `tic` from `ncurses` distribution, to prepare `TERMINFO` for a VT220.

Installing `ncurses` is a multiple step process which involves ensuring that the required build utilities are already installed. Once these utilities are installed, you can compile or build `ncurses`.

For more information on the `ncurses` and the build utility versions see, [Chapter 2, "System Requirements"](#).

11.2.2 Installing `libiconv` on HP-UX Itanium

The `libiconv` libraries for HP-UX Itanium B.11.23 need to be installed for running VT220 on the HP-UX Itanium platform.

Note: `Libiconv` library has run-time dependencies on `libgcc` and `gettext`. Therefore, you must install these packages while installing `libiconv`.

11.2.3 Installing the Sterling Multi-Channel Fulfillment Solution Mobile Application on VT220 Mobile Terminals

The Sterling Multi-Channel Fulfillment Solution Mobile Application can be installed on AIX, HP-UX, Linux, or Solaris operating systems. The VT220 terminal emulation software is installed along with the Sterling Multi-Channel Fulfillment Solution Mobile Application as described in this document for the respective operating systems.

To install the Sterling Multi-Channel Fulfillment Solution Mobile Application for VT220 terminal emulation:

1. The Sterling Multi-Channel Fulfillment Solution Mobile Application for VT220 terminal emulation is installed under the directory `<INSTALL_DIR>/mobileapp/vt220/<OS-name>`; where `<OS-name>` is the applicable operating system from the following: `aix`, `hpux`, `linux`, or `solaris`. This installation location is referenced as `<VT220_HOME>`.

2. Grant execute permission to `<VT220_HOME>/yantramobileapp` and `<VT220_HOME>/keyseq`.
3. Set up the VT220 emulation terminal as described in "[Setting Up the Terminal](#)" on page 140. Using the `<VT220_HOME>/keyseq` binary, verify that the keys F1 through F12 display the respective keys on the `keyseq` program output.

Setting Up the Terminal

When setting up a terminal for use with the Sterling Multi-Channel Fulfillment Solution Mobile Application for VT220 terminal emulation, you must perform the following preliminary actions:

1. Set your environment variable `TERM` to `vt220`.
2. Under the `<VT220_HOME>` directory, create a directory called `terminfo`.
3. From the `terminfo` directory use the `infocmp` command to define your terminal information as:

```
$<ncurses_home>/infocmp > vt220.ti,
```

where `<ncurses_home>` is the ncurses binaries installation location.

Note: If an installation location is not specified during the ncurses installation, the ncurses binaries are installed in the `/usr/bin` folder.

The `infocmp` command de-compiles the terminal information, and the resulting file can be edited to map the keystrokes observed by running `keyseq`.

4. You must define a `TERMINFO` variable (if you do not already have one) to tell the terminal where to find information on a particular terminal type. On BASH type systems, this is done using the following command:

```
$ export TERMINFO=<path to some directory that contains  
the .ti file>
```

5. Now run the ncurses `tic` command as follows to compile your newly built terminal information file:

```
$ <ncurses_home>/tic vt220.ti
```

where <ncurses_home> is the ncurses binaries installation location.

Note: If an installation location is not specified during the ncurses installation, the ncurses binaries are installed in the /usr/local/bin folder.

The `tic` command places the compiled version in the appropriate place under the `$TERMINFO` directory.

6. To ensure that all function keys are properly mapped, use the `<VT220_HOME>/keyseq` program. This shows you what key sequence is returned when a key is pressed. Run it and press the function keys when prompted.

For example, if the F1 key is pressed and:

Press a key (Return to end): Key Value returned: 27

Press a key (Return to end): Key Value returned: 91

Press a key (Return to end): Key Value returned: 49

Press a key (Return to end): Key Value returned: 49

Press a key (Return to end): Key Value returned: 126

is printed to STDOUT. These values are decimal values.

Convert these values to their HEX equivalents. Then, using the "Hexadecimal - character" set from the `man ascii` command on UNIX, edit the `vt220.ti` file created above. Use [Table 11–2, "Terminal Information - Common Keys and Codes"](#) on page 142 to decide which values to edit.

In the example above, the F1 key maps to `kf1` (from [Table 11–2](#)). Therefore, you must change the value of `kf1` in the `vt220.ti` file (generated in [Step 3](#)) to:

```
\E[11~
```

7. Repeat [Step 6](#) for all the keys you want to map.
8. Re-compile the edited `vt220.ti` file as directed in [Step 5](#).

9. Once you have edited and compiled your terminal information file, test the changes you made by running `keyseq` again. If all keys are properly defined, `keyseq` returns a string description and the numeric value of the key.

The most common keys and their codes in the terminal information file are in [Table 11–2](#):

Table 11–2 Terminal Information - Common Keys and Codes

Code	Key
kcub1	Left arrow
kcuf1	Right arrow
kcuu1	Up arrow
kcud1	Down arrow
kf1 – kf12	F1 – F12 keys

To launch the Sterling Multi-Channel Fulfillment Solution Mobile Application using the VT220 emulation terminal and access online help:

1. Set an environment variable `VT220_HOME` pointing to the folder containing the VT220 executable. This environment variable must be set in the shell from where the `yantramobileapp` executable is invoked.
2. To launch the application, type the following command in the operating system shell:
`yantramobileapp -i <ip_address> -p <port_number>`

12

Configuring Properties

Property files contain properties that control the operation of the Sterling Multi-Channel Fulfillment Solution. By modifying the values of these properties, you can customize the Sterling Multi-Channel Fulfillment Solution to suit your business and technical needs.

After installing the Sterling Multi-Channel Fulfillment Solution, most property and script files do not need any further configuration for basic operation of the system. However, if you want to customize any specific operations—for example, setting a different logging level—you will need to edit (and in some cases, create) certain property or .xml files.

In general, changes to properties are not made in the specific property files themselves; changes are made to the `customer_overrides.properties` file or `sandbox.cfg`.

Detailed information about each configurable property in `sandbox.cfg`, `yfs.properties.*`, `management.properties`, and `dbclassCache.properties` is located in [Appendix A, "Property Files Reference"](#).

Warning: Change only the properties included in this appendix. Sterling Commerce does not support changes to any other properties.

Note: This chapter contains the information required to complete [Step 11](#) indicated on the "[Installation Checklist](#)" on page 1.

Note: If you are upgrading from a prior release, see the *Sterling Multi-Channel Fulfillment Solution Upgrade Guide* that applies to your implementation **before** continuing with the set up of the Sterling Multi-Channel Fulfillment Solution. Changes have been made to the properties files in the Release 8.0, so you must read the Upgrade Guide for information about how this change impacts your system.

12.1 Initial Settings for Properties Files

In the Sterling Multi-Channel Fulfillment Solution, property files are generated when the `<INSTALL_DIR>/bin/setupfiles.sh` (or `setupfiles.cmd`) script is run from the corresponding initial settings files. The initial settings files are shipped with the product and are present in the same `<INSTALL_DIR>/properties` directory.

The `*.in` files contain variable placeholders (parameters) for properties that contain installation- or environment-specific information (such as database host, port, or credential information).

The installer gathers these parameters (either interactively or using a silent installation parameter file) and places this initial configuration information into a special parameters file called `sandbox.cfg`. Using the parameters contained in the `sandbox.cfg`, the `*.in` files are processed by the `setupfiles` script to create the final properties files that are used by the product during runtime.

Sterling Commerce does not recommend that you modify or change any properties in files ending with `.in` because newer versions or patches of the product will overwrite your changes. Sterling Commerce also does not recommend that you change a property file that has a corresponding `.in` file because the `setupfiles` script will re-create the properties file again, thus causing you to lose your changes. If you need to override or change the value of a property, see [Section 12.2, "Overriding Properties"](#).

12.2 Overriding Properties

Even though it is recommended that you do not change property files directly, situations do occur that require property changes. For these situations, use the `customer_overrides.properties` file, `sandbox.cfg` file, and `log4j.custom.xml` file, which are described below.

- `customer_overrides.properties` – If you need to change the value of a property that is not parameterized in the `.in` file(s), you can override that property by adding an entry for it to the `customer_overrides.properties` file. Making changes to this override file instead of individual property files ensures that none of your changes are lost in an upgrade or patch installation, and lets you see most of the overrides in your configuration at a glance, rather than having to check each property file individually. This file is not delivered with the product; you must create it the first time you have a need for it. For information about creating and using the `customer_overrides.properties` file, see [Section 12.2.2, "Using the Customer_Overrides.properties File"](#).
- `sandbox.cfg` – Contains name-value parameters that are merged with each `*.in` file to create the final properties files. For values that are parameterized in the `*.in` files, you can supply that parameter in the `sandbox.cfg` file. For information about using the `sandbox.cfg` file, see [Section 12.2.1, "Using the Property Parameters File \(sandbox.cfg\)"](#).
- `log4j.custom.xml` – For changes to logging properties, you create a file called `log4j.custom.xml`. You make changes to basic logging properties in this file. There are additional logging properties in the `yfs.properties.in` file. To make changes to these, you create entries in the `customer_overrides.properties` file. For information about creating and using the `log4j.custom.xml` file, see [Section 12.5, "Logging Configuration"](#).

Once you have read the sections on using `customer_overrides.properties` and `sandbox.cfg`, you may want to read the following sections, which give more detail about specific types of changes that are made using these two files:

- [Section 12.3, "Properties for LDAP User Authentication"](#)
- [Section 12.4, "Properties to Prevent Cross-Site Script Vulnerabilities"](#)
- [Section 12.5, "Logging Configuration"](#)

12.2.1 Using the Property Parameters File (sandbox.cfg)

The `<INSTALL_DIR>/properties/sandbox.cfg` file contains name-value parameters that are merged with each `*.in` file to create the final properties files. You can examine any `*.in` file and if you find a value that is parameterized, you can supply that parameter in the `sandbox.cfg` file. A parameter is contained within the `'&'` and `';'` characters. For example, the `jdbc.properties.in` file contains the following property:

```
oraclePool.user=&ORA_USER;
```

The `&ORA_USER;` signifies a parameter. If the `sandbox.cfg` file contains the entry: `ORA_USER=oracle`, the resulting `jdbc.properties` file will contain the following property:

```
oraclePool.user= oracle
```

You can edit the information in the `sandbox.cfg` file at any time to change values that have been created by the installer or to reflect changed setup parameters.

Most of the parameters in the `sandbox.cfg` file and the `*.in` files are not used at runtime by the product. Consequently, if you change a parameter in the `sandbox.cfg` file, you must run the `setupfiles` script so that the runtime property files are re-created with the updated values.

See [Appendix A, "Property Files Reference"](#), for a list of properties that can be changed and a description of each.

12.2.2 Using the Customer_Overrides.properties File

The `sandbox.cfg` parameters let you change the properties defined with variables. However, if you need to change the value of a property that is not parameterized in the `.in` file(s), you can override that property by creating a special file called `customer_overrides.properties`.

For each property that you want to override, you must have the following information:

- `PROPERTY_FILE_NAME_PREFIX` - Name used in the `servers.properties` file to reference the actual property file.

- **PROPERTY_NAME** - The name of the property as used in the specified property file.
- **PROPERTY_VALUE** - The value you want to assign to the property.

These values are used to create an entry in `customer_overrides.properties` that looks similar to the following example:

```
PROPERTY_FILE_NAME_PREFIX.PROPERTY_NAME=PROPERTY_VALUE
```

12.2.2.1 Locating the Property File Name Prefix

To find the `PROPERTY_FILE_NAME_PREFIX` for a property file:

1. In the `<INSTALL_DIR>/properties` directory, locate the `servers.properties` file and open it in a text editor.
2. Find the entry for the desired property file.
3. The part of the entry before the equal sign (=) is the prefix that you will use in `customer_overrides.properties`. Make note of it.

For example, locate the entry for `jdbc.properties` in `servers.properties`:

```
jdbcService=<INSTALL_DIR>/properties/jdbc.properties
```

`jdbcService` is the prefix for the `jdbc.properties` property file.

To create an entry in the `customer_overrides.properties` file:

1. In the `<INSTALL_DIR>/properties` directory, locate (or create if necessary) the `customer_overrides.properties` file.
2. Open the `customer_overrides.properties` file in a text editor.
3. Add the properties that you want to override, using the following format:

```
PROPERTY_FILE_NAME_PREFIX.PROPERTY_NAME=PROPERTY_VALUE
```

4. Save and close the `customer_overrides.properties` file.
5. Stop your Sterling Multi-Channel Fulfillment Solution server.
6. Rebuild the EAR, following the instructions for your application server type as described in [Chapter 14, "Deploying the Sterling Multi-Channel Fulfillment Solution"](#).

7. Restart your Sterling Multi-Channel Fulfillment Solution server and all agent and integration servers.

12.2.2.2 Example 1: Overriding a property from yfs.properties.in

1. Check servers.properties to find the PROPERTY_FILE_NAME_PREFIX. In server.properties, the line for yfs.properties.in is:

```
yfs=<INSTALL_DIR>/properties/yfs.properties
```

yfs is the PROPERTY_FILE_NAME_PREFIX.

2. The yfs.properties file contains a property called yfs.smtp.connectionpool.enable. For an override entry, the PROPERTY_NAME value would be yfs.smtp.connectionpool.enable. So far, the entry for customer_overrides.properties so far would be:

```
yfs.yfs.smtp.connectionpool.enable
```

3. The default value for yfs.smtp.connectionpool.enable is "true." To override this default, you would add the value of "false" to the entry. The completed override entry for the example is shown below.

```
yfs.yfs.smtp.connectionpool.enable=false
```

Note: For more information about properties in the yfs.properties files that can be changed, see [Appendix A, "Property Files Reference"](#).

12.2.2.3 Example 2: Overriding a property from dbclassCache.properties

1. Check servers.properties to find the PROPERTY_FILE_NAME_PREFIX. In server.properties, the line for dbclassCache.properties says:

```
dbclassCache=<INSTALL_DIR>/properties/dbclassCache.properties
```

dbclassCache is the PROPERTY_FILE_NAME_PREFIX.

2. The dbclassCache.properties file contains a property called sci.globalcache.object.size. For an override entry, the PROPERTY_NAME value would be sci.globalcache.object.size. So far, the entry for customer_overrides.properties so far would be:

```
dbclassCache.sci.globalcache.object.size
```

3. The default value for `sci.globalcache.object.size` is 10000. To override this default, you would add the new value (in this example, 15000) to the entry. The completed override entry for the example is shown below.

```
dbclassCache.sci.globalcache.object.size=15000
```

Note: For more information about properties in the `dbclassCache.properties` file that can be changed, see [Appendix A, "Property Files Reference"](#).

Note: You do not need to run the `setupfiles` script after making a change to the `customer_overrides.properties` file, because no parameters are being replaced. However, you do need to stop and restart your Sterling Multi-Channel Fulfillment Solution server and all agent and integration servers for the changes to take effect.

12.3 Properties for LDAP User Authentication

This section assumes you understand how LDAP servers work. Sterling Commerce also recommends that you read the following documents on LDAP technology:

- W. Yeong, T. Howes, and S. Kille, *RFC 1777 - Lightweight Directory Access Protocol*. March 1995. Available at <http://rfc.sunsite.dk/rfc/rfc1777.html>.
- Mark Wilcox, *Implementing LDAP*. Wrox Press, 1999.

By default, all authentication is performed against the Sterling Multi-Channel Fulfillment Solution database. When a user enters a login ID and password, it is validated against the login ID and password that is stored in the database. This requires the administrator of the Sterling Multi-Channel Fulfillment Solution system to set up login IDs and passwords for each user.

Alternatively, the Sterling Multi-Channel Fulfillment Solution Consoles support LDAP-based user authentication. You may choose to use an LDAP server for authentication. When using LDAP, the users, user groups, and

access control must be set up in the Sterling Multi-Channel Fulfillment Solution system.

The Sterling Multi-Channel Fulfillment Solution also supports password expiration through LDAP. Your custom code for user authentication is interfaced with the Sterling Multi-Channel Fulfillment Solution authentication mechanism. If your custom code contains `ExpireInDays` with a numeric value of `<X>`, then a message to reset the password appears in the Sterling Multi-Channel Fulfillment Solution home page. If the map contains `ChangePasswordLink` then the message contains a link to the location specified. Clicking on the link opens a new window with the given `ChangePasswordLink`.

Since the various implementations of LDAP, handle password expiration differently a sample `YFSLDAPAuthenticator` is modified to provide an example of one particular implementation. This is located in the `<INSTALL_DIR>/xapidocs/code_examples/java` directory.

To set properties for LDAP-based authentication:

1. Install the LDAP server (see the installation instructions from your LDAP server vendor).
2. If a JAAS-compliant provider is used, create a JAAS configuration file with the following lines:

```
LDAP
{
    // refer to the JAAS compliant service provider for the login
    module details.
    <Class Name of the Login Module as specified by the Security
    provider> required
    debug=true;
};
```

3. In your `customer_overrides.properties` file, specify the LDAP properties described in [Table 12–1, "LDAP-Based Authentication Properties"](#). For more information about the `customer_overrides.properties` file, see [Section 12.2, "Overriding Properties"](#).

Table 12–1 LDAP-Based Authentication Properties

Property	Description
In the <code>customer_overrides.properties</code> file, specify:	
<code>yfs.yfs.security.authenticator</code>	Develop a new class that implements the <code>com.yantra.yfs.japi.util.YFSAuthenticator</code> interface and set the new classname as value for this property. Note: The Sterling Multi-Channel Fulfillment Solution provides a sample <code>com.yantra.yfs.util.YFSLdapAuthenticator</code> class that you can use for reference.
<code>yfs.yfs.security.ldap.factory</code>	If the default implementation is used, this property specifies the LDAP context factory classname as in your LDAP Server configuration. Set this property value to <code>com.sun.jndi.ldap.LdapCtxFactory</code> .
<code>yfs.yfs.security.ldap.url</code>	If the default implementation is used, this property specifies the URL used to access your LDAP Server. For example, <code>yfs.security.ldap.url=ldap://MyServer:800</code> .
<code>yfs.yfs.security.ldap.o</code>	If the default implementation is used, this property specifies the Sterling Multi-Channel Fulfillment Solution organization in your LDAP Server configuration.
<code>yfs.yfs.security.ldap.ou</code>	If the default implementation is used, this property specifies the Sterling Multi-Channel Fulfillment Solution organizational unit in your LDAP Server configuration.
<code>yfs.yfs.jaas.loginmodule</code>	If using JAAS, set this property value to <code>LDAP</code> .
<code>yfs.yfs.security.authenticator</code>	If using JASS, set this property value to <code>com.yantra.interop.services.security</code> .
WebLogic startWLS startup file	
<code>-Djava.security.auth.login.config</code>	If you are using JAAS and WebLogic, specify the full path to your JAAS configuration file.
In the Configurator UI	
Configure organizations, organization units, and users.	All the users who need to access the Sterling Multi-Channel Fulfillment Solution system must be set up under the LDAP server. All Sterling Multi-Channel Fulfillment Solution users must belong to the same organizational unit.

12.4 Properties to Prevent Cross-Site Script Vulnerabilities

In some cases, data to and from the Sterling Multi-Channel Fulfillment Solution can contain HTML characters that impact the display and the original intent of the input. In addition, data can be input that contains malicious HTML, such as commands embedded within `<SCRIPT>`, `<OBJECT>`, `<APPLET>`, and `<EMBED>` tags.

The `yfs.htmlencoding.triggers` property in the `yfs.properties.in` file specifies the following characters:

- Greater than symbol ("`>`")
- Less than symbol ("`<`")
- Right parenthesis ("`)`")
- Right bracket ("`]`")

If needed, you can add any other characters necessary for your specific implementation by using the `customer_overrides.properties` file. For more information about the `customer_overrides.properties` file, see [Section 12.2, "Overriding Properties"](#).

The presence of any of the characters specified triggers the Sterling Multi-Channel Fulfillment Solution to encode the data.

For more detailed information about malicious scripts, see the following articles:

- CERT Advisory, *Malicious HTML Tags Embedded in Client Web Requests*. Available from <http://www.cert.org/advisories/CA-2000-02.html>.
- CERT Advisory, *Frequently Asked Questions About Malicious Web Scripts Redirected by Web Sites*. Available from http://www.cert.org/tech_tips/malicious_code_FAQ.html.

12.5 Logging Configuration

The Sterling Multi-Channel Fulfillment Solution includes basic logging functionality. However, you can change logging parameters, if necessary, to better suit your needs.

Warning: Before setting up the logging parameters, ensure that you understand the log4j utility. For detailed information about this utility, see <http://jakarta.apache.org/log4j>.

To set properties for logging:

To use a log4j file inside the EAR:

1. Create new file in <INSTALL_DIR>/resources, for example log4jconfig.custom.xml.
2. Modify customer_overrides.properties used by your application server to add yfs.log4j.configuration=/resources/log4jconfig.custom.xml.
3. Run <INSTALL_DIR>/bin/deployer.sh -t resourcejar. This puts log4jconfig.custom.xml in resources.jar.
4. Rebuild and redeploy the EAR, following the instructions for your application server type as described in [Chapter 14, "Deploying the Sterling Multi-Channel Fulfillment Solution"](#).

To use a log4j file outside of the EAR:

1. Create new file in your classpath, for example /folder/log4jconfig.custom.xml.
2. Modify customer_overrides.properties used by your application server to add yfs.log4j.configuration=/folder/log4jconfig.custom.xml. Note that the "/" in front of the parameter is required.
3. Restart your application server.

Table 12–2 Logging Properties

Property	Description
In the log4j configuration XML File	
<priority> subelement of the <root> element	<p>Specify the level of logging desired. Sterling Commerce recommends that you set the value of this attribute to ERROR.</p> <p>The following are valid values for logging levels:</p> <ul style="list-style-type: none">• ERRORDTL• ERROR• WARN• INFO• TIMER• SQLDEBUG• DEBUG• VERBOSE
<appender> subelement	<p>At the root level, this attribute specifies the associated name and class attribute. Choose a valid log4j appender class.</p> <p>Each subelement can also specify the layout of the message through the <layout> subelement and can filter for levels through the <filter> subelement.</p> <p>Instead of hardcoding the absolute path for the log file under the appender you plan to use, Sterling Commerce recommends that customers should also use a <code>\${LOG_DIRECTORY}</code> parameter in the <code>log4jconfig.xml</code> file and invoke the JVM with a <code>-DLOG_DIRECTORY=<application_log_directory>/<logFileName></code> option.</p>

12.6 Enabling Different Properties for Individual Processes

It is possible to specify different properties for each process you are running. To do this, you must have a different `servers.properties` and `customer_overrides.properties` file for each process that you are running. In the start scripts for the process, set your `-DvendorFile=<your custom servers.properties>`. In your `customer servers.properties`, change the entry for `customer_overrides.properties` to point to your new `customer_overrides.properties`. See [“Using the](#)

[Customer_Overrides.properties File](#) on page 146 for details on how to override a specific property.

13

Configuring Utilities

The Sterling Multi-Channel Fulfillment Solution provides script files (.sh for UNIX and .cmd for Windows) that you must customize using the directions provided in this chapter.

This chapter describes all the utilities supplied by the Sterling Multi-Channel Fulfillment Solution, organized in the order in which you are likely to use them. It describes generic customizations that apply to most or all utilities. Further details specific to each utility are provided throughout the rest of this guide.

This chapter also provides instructions required to complete [Step 12](#) indicated on the [Installation Checklist](#).

13.1 Installation Utilities

Installation utilities enable you to install the Sterling Multi-Channel Fulfillment Solution. These utilities are present in the `<INSTALL_DIR>/bin` directory. Some of the utilities used for installing the various configurations of the Sterling Multi-Channel Fulfillment Solution are `"loadDefaults"` and `"dbverify"`.

loadDefaults

This utility loads the standard installation database configuration, known as the "factory defaults". For detailed information, see [Section 13.1.1.3, "Loading the Sterling Multi-Channel Fulfillment Solution Database Factory Defaults After Installation"](#).

dbverify

This utility performs database schema creation, verification, and correction. Dbverify is used to ensure database schema integrity. When run, it invokes a Java class to compare a database with the entity XMLs and generates the SQL statements that would make the database match the entity repository and generates SQL statements for any differences between the two.

During the installation process, dbverify is used to generate SQL scripts to create the database schema or tables and indexes. These SQL statements are then run against the database, unless you choose to manually create database schemas after installation. For more information about manually creating the database schemas after installation, see [Section 13.1.1, "Creating Database Schemas and Loading Factory Defaults After Installation"](#). For detailed information about setting up and running dbverify, see [Section 13.1.1.2, "Verifying the Database Schema"](#).

install3rdparty

This utility copies supplied resources into the Sterling Multi-Channel Fulfillment Solution directory structure, and can append added jar files to the global classpath, agent classpath, or application server EAR file.

installService

This utility installs programs for specific tasks, like a regression test jar file or a patch jar file.

setupfiles

This utility checks the various initial product settings files (*.in) files for variables and updates the corresponding files with the values defined in the `sandbox.cfg` file to create the final files that are used by the product during runtime.

13.1.1 Creating Database Schemas and Loading Factory Defaults After Installation

By default, the database schemas are created and factory defaults are automatically loaded during installation. However, you can tell the installation process to skip these tasks, then perform the tasks manually after installation. To have the installation process skip creating the

database schemas and loading factory defaults, do one of the following, depending on the type of installation you choose:

- If you are using the GUI Installer on UNIX/Linux or on Windows, check the *Do you wish to upgrade from a previous Install? Also check this option if you do NOT want Database DDLs and FactorySetup installed* option. For more information about the GUI installation process, see [Section 7.3.2, "Running the Installation Program in UNIX or Linux \(GUI-Based\)"](#) or [Section 8.3.1, "Running the Installation Program in Windows \(GUI-Based\)"](#).
- If you are using the text-based installation process on UNIX/Linux, answer "yes" to the question "Do you wish to upgrade from a previous Install? Also check this option if you do NOT want Database DDLs and FactorySetup installed." For more information about the text-based installation process, see [Section 7.3.3, "Running the Installation Program in UNIX or Linux \(Text-Based\)"](#).
- If you are using the silent installation method, set the `LOAD_FACTORY_SETUP` parameter in your silent installation file to false prior to running the installation and use the `-cluster` option as part of the installation command. For more information about silent installation, see [Section 7.2.7, "About Silent Installations \(UNIX/Linux\)"](#) or [Section 8.2.6, "About Silent Installations \(Windows\)"](#).

13.1.1.1 Updating Properties Files After Installation

After installing the Sterling Multi-Channel Fulfillment Solution in Upgrade mode, reset the following properties in `<INSTALL_DIR>/properties/sandbox.cfg` as shown here:

- `REINIT_DB=true`
- `LOAD_FACTORY_SETUP=true`
- `NO_DBVERIFY=false`
- `DB_SCHEMA_OWNER=<your database schema owner>`

After setting the properties, you must re-run `setupfiles.sh/cmd` from the `<INSTALL_DIR>/bin` folder.

For more information about editing the `sandbox.cfg` file, see [Chapter 12, "Configuring Properties"](#).

13.1.1.2 Verifying the Database Schema

To use the dbverify utility to verify the database schema:

1. From the command line, run the dbverify script from the <INSTALL_DIR>/bin/ folder as follows:

dbverify.sh (on UNIX and Linux)
or
dbverify.cmd (on Windows)
2. If you have enabled the text search feature, edit the <INSTALL_DIR>/bin/dbverify.sh (or .cmd on Windows) script. For MS SQL and DB2 databases only, add the -DBNAME <database_name> parameter as mentioned below. For example,

```
%JAVA_HOME%\bin\java com.yantra.tools.dbverify.DbVerifyCommandLine -b
%<INSTALL_DIR>% -DBNAME %<database_name>% -u %USERNAME% -p %PASSWD% -d
%DRIVER% -url %URL% -g Y -DT
%<INSTALL_DIR>%/repository/datatypes/datatypes.xml
```

Where <database_name> refers to the name of the database for which text search feature is enabled. The "-DBNAME" parameter is required only for MS SQL and DB2 databases.

And add the "-TSIT" parameter as mentioned below. For example:

```
%JAVA_HOME%\bin\java com.yantra.tools.dbverify.DbVerifyCommandLine -b
%<INSTALL_DIR>% -u %USERNAME% -p %PASSWD% -d %DRIVER% -url %URL% -g Y
-DT %<INSTALL_DIR>%/repository/datatypes/datatypes.xml -TSIT
<tsindex_type>
```

Where <tsindex_type> refers to the type of text search indexes you have created such as ctxcat or context. The "-TSIT" parameter is required only for Oracle database.

Note: If you change the text search index type in Oracle from ctxcat to context or vice-versa, the updated create and drop SQL scripts can be found in the <INSTALL_DIR>/bin/EFrame_TextIndexUpdates.sql file.

3. The differences between the entity XMLs and the database are generated in the form of SQL scripts, which can be run against the database to rectify the differences.

For example, if there is a mismatch in the size of a datatype for a column [varchar2(20) to varchar2(40)] that has an associated index, dbverify generates SQL statements for:

- Dropping the Index
- Changing the size of the datatype for the column
- Creating the new Index

The three SQL statements described in the previous list appear in different *.sql files. The appropriate *.sql files must be run in the proper order as follows:

- a. Run the <INSTALL_DIR>/bin/EFrame_IndexDrops.sql for dropping the index.
- b. Run the <INSTALL_DIR>/bin/EFrame_TableChanges.sql for altering the size of the datatype for a column.
- c. Run the <INSTALL_DIR>/bin/EFrame_IndexAdds.sql for creating a new index.

If the SQL statements are not run in the sequence as mentioned above, it results in script failure.

The following scripts are generated:

Note: All scripts listed below can be found in the <INSTALL_DIR>/bin directory.

Script Name	Description of the script
EFrame_Sequence.sql	<p>Contains all the additional sequences that need to be created.</p> <p>Note: If you are using an MS SQL Server 2000/2005 database, the EFrame_Sequence.sql script is not created when you run the dbverify command.</p>

Script Name	Description of the script
EFrame_TableChanges.sql	Contains all the table column differences that need to be applied on the database schema. Modify this file to reference your tablespaces.
EFrame_Drops.lst	This list is provided as an example of tables that can be removed. Note: The Sterling Multi-Channel Fulfillment Solution does not provide a .sql file for removing tables from the database.
EFrame_IndexAdds.sql	Adds all of the indexes that need to be created in the database. Modify this file to reference your tablespaces.
EFrame_IndexDrops.sql	Removes any extra indexes in the database.
EFrame_TextIndexAdds.sql	Adds new text search indexes that need to be created in the database.
EFrame_TextIndexDrops.sql	Removes text search indexes from the database.
EFrame_TextIndexModify.sql	Updates the text search indexes in the database.
EFrame_TextIndexUpdates.sql	When executed, updates the content of the text indexes.
EFrame_UpdateQueries.sql	For upgrades, updates the table column values in order to apply other changes made to the columns. For example, if a table column is changed from nullable to not nullable in the installation of a previous release, the column values must be updated before the column can be made not null in the current release because the column default values for the current release may contain null values.

Note: The <INSTALL_DIR>/bin/EFrame_Drops.lst indicates extra objects in the database. These extra objects could be custom objects or objects that are dropped as the result of a schema change or an upgrade. Please look through this script carefully.

This script may also contain reduced columns. These are columns that were changed to have a smaller size in the newer version. These changes are suppressed because:

- Not all databases will allow you to apply the changes.
- Databases that do allow you to apply the changes can behave unpredictably if the table already contains values that are longer than the new length.

4. Run the scripts specified for your database type, as shown in the following lists.

Oracle

For Oracle, run:

```
<INSTALL_DIR>/database/oracle/scripts/yfs_master_db_script.  
sql
```

Running this command executes the following scripts in <INSTALL_DIR>/database/oracle/scripts directory:

- yfs_view_custom.sql
- CustomDBViews/Interop_Views.sql
- CustomDBViews/ImportExport_View.sql
- CustomDBViews/yfs_cross_reference_vw.sql
- CustomDBViews/yfs_order_release_line.sql
- CustomDBViews/yfs_order_release_line_vw.sql
- CustomDBViews/yfs_invtdmddtl_vw.sql
- CustomDBViews/InvSnapshot_vw.sql
- CustomDBViews/yfs_iba_ord_demand_vw.sql

- CustomDBViews/yfs_iba_resv_demand_vw.sql

SQL Server 2000/SQL Server 2005

For SQL Server 2000 or SQL Server 2005, run:

```
<INSTALL_DIR>\database\sqlserver\scripts\yfssqlserver_master_db_script.cmd
```

Running this script executes the following scripts in the <INSTALL_DIR>\database\sqlserver\scripts directory:

- yfs_seq_sqlserver.sql
- CustomDBViews\yfs_invtdmddtl_vw.sql
- CustomDBViews\yfs_order_release_line.sql
- CustomDBViews\yfs_cross_reference_vw.sql
- CustomDBViews\yfs_order_release_line_vw.sql
- CustomDBViews\InvSnapshot_vw.sql
- CustomDBViews\yfs_iba_ord_demand_vw.sql
- CustomDBViews\yfs_iba_resv_demand_vw.sql
- CustomDBViews\ImportExport_View.sql
- CustomDBViews\Interop_Views.sql

DB2

For DB2, run the following scripts in the <INSTALL_DIR>/database/db2/scripts directory individually:

- CustomDBViews/yfs_invtdmddtl_vw.sql
- CustomDBViews/yfs_order_release_line.sql
- CustomDBViews/yfs_cross_reference_vw.sql
- CustomDBViews/yfs_order_release_line_vw.sql
- CustomDBViews/InvSnapshot_vw.sql
- CustomDBViews/yfs_iba_ord_demand_vw.sql
- CustomDBViews/yfs_iba_resv_demand_vw.sql
- CustomDBViews/ImportExport_View.sql

- CustomDBViews/Interop_Views.sql

13.1.1.3 Loading the Sterling Multi-Channel Fulfillment Solution Database Factory Defaults After Installation

To load the Sterling Multi-Channel Fulfillment Solution database factory defaults after the product installation, load the defaults using the script applicable to your operating system. From the command line, run the `<INSTALL_DIR>/bin/loadDefaults.sh` command on UNIX and Linux or the `<INSTALL_DIR>\bin\loadDefaults.cmd` command on Windows and pass the absolute file path to the installer xml file and the absolute path to the factory setup XMLs:

On UNIX/Linux:

```
loadDefaults.sh
<INSTALL_DIR>/installed_data/sscap/components/complete_installation/factorysetup
/installer.xml
<INSTALL_DIR>/installed_data/sscap/components/complete_installation/factorysetup
/XMLS
```

On Windows:

```
loadDefaults.cmd
<INSTALL_DIR>\installed_data\sscap\components\complete_installation\factorysetup
\installer.xml
<INSTALL_DIR>\installed_data\sscap\components\complete_installation\factorysetup
\xMLS
```

Note: If the factory default installation stops before it is finished, a file named "installer.xml.restart" is created. This file records the location where the installation was stopped, and it is used the next time the factory defaults are installed.

You can also generate audits when running loadDefaults script by overriding the value of the AUDIT_LOAD_DEFAULTS property and setting it to true. By default, this property is set to false. To override the value of this property add an entry for it in the `sandbox.cfg` file. For more information about modifying properties and `sandbox.cfg` file, see [Appendix A.5, "Sandbox.cfg"](#).

13.1.2 Installing Third-Party JAR Files

You can use the `install3rdParty` utility to add third-party custom jars to the classpath of various utilities and enterprise archive (EAR) files.

For example, to install the `wlclient.jar`, use the following command:

```
<INSTALL_DIR>/bin/install3rdParty.sh weblogic 10 -j
<BEA_HOME>/wlserver_10.0/server/lib/wlclient.jar -targetJVM AGENT
```

This command causes the `wlclient.jar` file to be copied from the WebLogic installation location into the product installation location (`<INSTALL_DIR>/jar/wellogic/10/wlclient.jar`). The utility then updates the

`<INSTALL_DIR>/properties/AGENTDynamicclasspath.cfg.in` file with the new jar file and invokes the `<INSTALL_DIR>/bin/setupfiles.sh` utility to regenerate the `AGENTDynamicclasspath.cfg` file from the modified ".in" file.

Note: If you want to make this new JAR available to the Application Server and Agents when running the `install3rdParty` utility, based on your requirement pass the following arguments:

- APP—If you want to add the new JAR to the EAR file.
- AGENT—If you want to add the new JAR to the Agent and Integration Servers dynamic classpath.

Keep the following in mind when using the `install3rdParty` utility to update a classpath:

- The order of lines in the dynamic classpath files determine the order of the classpath for the application server or agent.
- Whatever is in the beginning of the file is analogous to the jar being in the beginning of the classpath.

For help in using `install3rdParty`, enter the command, including the `-help` option, on the command line. The `install3rdParty` utility prints a usage message.

Note: Sterling Multi-Channel Fulfillment Solution supports only APP and AGENT options for the install3rdParty utility. All other options that are being displayed with the -help message command are not supported.

13.2 Upgrade Utilities

Upgrade utilities enable you to upgrade the Sterling Multi-Channel Fulfillment Solution. The upgrade-related utilities are described in depth in the *Sterling Multi-Channel Fulfillment Solution Upgrade Guide*.

MigrationValidator

For more information, see the *Sterling Multi-Channel Fulfillment Solution Upgrade Guide*.

migrator

The Data Migrator migrates your data. It has a properties file and takes in an XML file as a option.

13.2.1 Data Migrator

The Sterling Multi-Channel Fulfillment Solution provides a Data Migrator to ease the aspects of application configuration when loading factory defaults.

The Data Migrator reads in a task definition file which is passed to it as a parameter. These files are located in the <INSTALL_DIR>/Migration subfolders.

13.2.1.1 Data Migrator Task Definition Files

The Sterling Multi-Channel Fulfillment Solution installation provides a task definition file that is run by the Data Migrator.

A typical task definition has the following structure:

```
<Task Class="" When="">
<TaskInfo Completed="N">
...Task specific information...
</TaskInfo>
<ChildTasks>
```

```
<Task/>
</ChildTasks>
</Task>
```

The important elements and attributes in the task definition file are described in the following sections for informational purposes only. Do not modify the task definition file.

Task Element

The `<Task>` element identifies one task to be processed by the Data Migrator. The attributes of the `<Task>` element are:

- **Class** - Attribute for specifying the name of the Java Class responsible to process this task. While processing this task, the Data Migrator invokes a pre-defined method in this Class and passes it the `TaskInfo` element as a parameter to the method. This element can be used by the Class to store and retrieve task-specific information. It is not possible to write your own implementation for this class.
- **When** - Attribute that sequences a group of related tasks. Valid values are:
 - **First** - Value that specifies which task to invoke first, before processing any related tasks (defined in the `ChildTasks` element). If a task defined as **First** fails, the related tasks defined under `ChildTasks` are not processed.
 - **AfterChildren** - Value that specifies a task is to be processed only *after* all tasks defined under `ChildTasks` have been successfully processed.

TaskInfo Element

The `<TaskInfo>` element contains task specific information and is dependent on the implementation of the class responsible for the task.

For example, Sterling Multi-Channel Fulfillment Solution uses a class called the `XMLMigrator` to load and configure data into tables from XML files. This class stores a list of Entities that it needs to load into the database under the `TaskInfo` object. This class is used extensively in the task definition files shipped with the software.

ChildTasks Element

The <ChildTasks> element contains a list of tasks that need to be sequenced (either before or after) with respect to the containing task.

13.2.1.2 Data Migrator Restart File

While processing tasks defined in the input file, the Data Migrator periodically writes status information to a restart file in the same directory as the input task definition file. This file enables the Data Migrator to recover from incomplete runs without the need to process all defined tasks again.

If one or more tasks fail, or if the Data Migrator terminates abnormally, this file is used to complete the remaining tasks. In most cases, this file is used just for performance enhancement and individual tasks can be rerun.

It is strongly recommended that you preserve restart files across runs of the Data Migrator.

13.2.1.3 Data Migrator Log Files

Data Migrator data-related errors or warnings are logged in the error file configured in the `log4jconfig.xml` file. These errors usually result from business validation failures or contain instructions about data that the Data Migrator could not automatically process and requires manual correction. All the other errors are output to the Sterling Multi-Channel Fulfillment Solution log files.

13.3 Development Utilities

Development utilities enable you to customize the Sterling Multi-Channel Fulfillment Solution to suit your business needs. They are for use while running the Sterling Multi-Channel Fulfillment Solution in development mode.

Configuration Deployment Tool

The Configuration Deployment Tool enables you to migrate configuration data from your development environment to your production environment. For more information about the configuration deployment tool, see the *Sterling Multi-Channel Fulfillment Solution Configuration Deployment Tool Guide*.

Transaction Data Truncation Tool

When deploying the Sterling Multi-Channel Fulfillment Solution to a production environment, you may not want to include all of your transaction data. The Sterling Multi-Channel Fulfillment Solution provides a utility through which you can generate a script to remove transaction data prior to moving into your production environment.

To truncate transaction data:

1. From the `<INSTALL_DIR>/bin` directory use the following command appropriate for your database:

For Oracle and SQL Server:

```
./ant.sh(cmd) -f generateTruncateTransactionData.xml
```

For DB2:

```
./ant.sh(cmd) -Ddbtype=DB2 -f  
generateTruncateTransactionData.xml
```

2. The `TruncateTransactionTables.sql` script is generated and placed in the current directory.
3. To truncate your transaction data, run the newly generated `TruncateTransactionTables.sql` script against your database.

13.4 Runtime Utilities

These utilities start processes that run in the background. The setup of these utilities is described in detail in [Chapter 12, "Configuring Properties"](#).

Integration Server

An integration Server is a process that manages asynchronous services, such as messages to and from external systems. You can run the integration server using the

`<INSTALL_DIR>/bin/startIntegrationServer` script.

The Sterling Multi-Channel Fulfillment Solution Integration Server allows the Sterling Multi-Channel Fulfillment Solution to collaborate with different systems, organizations, and businesses all through a standard, uniform interface to all systems. The Sterling Multi-Channel Fulfillment

Solution Integration Server runs in its own Java Virtual Machine (JVM) environment, separate from your application server.

Agent Server

The agent server utility starts processes responsible for processing transactions generated by the time-triggered transactions (agents). You can start multiple instances of an agent server using the `<INSTALL_DIR>/bin/agentserver.sh <server_name>` script as many times as needed.

Trigger Agent

The trigger agent utility is used for scheduling time-triggered transactions.

You can override the agent criteria attributes only in the Real-time Availability Monitor. The command for triggering the Real-time Availability Monitor with override abilities is:

```
triggeragent.sh <criteriaID> -<AgentCriteriaAttribute>
<OverriddenValue> (or .cmd on Windows)
```

To enable this override, you should pass the `AgentCriteriaAttribute` and `OverriddenValue` as additional parameters to the java class in the `triggeragent.sh` (or `.cmd` on Windows) file as follows:

```
java com.yantra.ycp.agent.server.YCPAgentTrigger -criteria %*
```

Therefore, when you invoke:

```
triggerAgent.sh CustomCriteria -MyOverriddenParam DynamicValue
```

all the values are passed to the java class.

However, do not modify the parameters passed to the java class in the default `triggeragent.sh` (or `.cmd` on Windows) file. Make these changes in the file that you have copied and renamed from the `triggeragent.sh` (or `.cmd` on Windows). Also, the agent criteria XML code must have the `AllowedOverriddenCriteria` flag set to `Y`.

13.4.1 Setting Up the Runtime Utilities

The `CLASSPATH` for the `startIntegrationServer`, `agentServer` and `triggerAgent` scripts must include certain jar files in order for them to be used on WebLogic, WebSphere MQ, or JBoss.

To include these jar files:

- If you are using WebLogic JMS, use the `<INSTALL_DIR>/bin/install3rdparty` script to install the `wlclient.jar`, `wls-api.jar`, and `wljmsclient.jar` files and include them in the `AGENTDynamicclasspath.cfg` dynamic classpath file.

Note: For more information about using the `install3rdparty` script, see [“Installing Third-Party JAR Files”](#) on page 166.

- If you are using WebSphere:
 1. Add `AGENT_JAVA_DEFINES` variable to the `<INSTALL_DIR>/properties/sandbox.cfg` file as follows:


```
AGENT_JAVA_DEFINES=-Djava.ext.dirs=<APPCCLIENT_HOME>/classes:<APPCCLIENT_HOME>/lib:<APPCCLIENT_HOME>/installedChannels:<APPCCLIENT_HOME>/lib/ext:<APPCCLIENT_HOME>/plugins:<APPCCLIENT_HOME>/java/jre/lib/ext:<APPCCLIENT_HOME>/java/jre/lib -Dserver.root=<APPCCLIENT_HOME>
-Dcom.ibm.CORBA.ConfigURL=file:<APPCCLIENT_HOME>/properties/sas.client.props
```
 2. Run the command `<INSTALL_DIR>/bin/setupfiles (.sh for UNIX or .cmd on Windows)`.
 3. Add a line to the agent script that sources the `<APPCCLIENT>/bin/setupClient.sh` script where `<APPCCLIENT>` refers to the WebSphere appclient installation directory. For example, on Unix:


```
<APPCCLIENT>/bin/setupClient.sh
```

on Windows:

```
<APPCCLIENT>/bin/setupClient.cmd
```
 4. If you are using a WebSphere application server for JNDI, install the WebSphere application client software of the same version as the WebSphere application server. Any script that accesses the JNDI directory must include the following line:


```
. ${WAS_HOME}/AppClient/bin/setupClient.sh
```
- If you are using JBoss JMS, no additional modifications are required.

- Do **not** include any `*ui.jar` files.
- If you have developed custom Java classes (user exits, event handlers, and so forth), see the section on "Including Custom Classes" for your application server in [Chapter 14, "Deploying the Sterling Multi-Channel Fulfillment Solution"](#).

Deploying the Sterling Multi-Channel Fulfillment Solution

After configuring the Sterling Multi-Channel Fulfillment Solution according to your business needs, deploy it into production based on your application server. This chapter describes how to deploy the Sterling Multi-Channel Fulfillment Solution on BEA WebLogic, IBM WebSphere, and JBoss. Deployment is part of the general path that you follow when installing and deploying the Sterling Multi-Channel Fulfillment Solution:

1. Installing the application server (JBoss, WebLogic, or WebSphere). Refer to the documentation for the application server.
2. Installing the Sterling Multi-Channel Fulfillment Solution. Refer to the installation information for the operating system (UNIX/Linux or Windows).
3. Building the Enterprise Archive (EAR).
4. Starting the application server.
5. Deploying the Sterling Multi-Channel Fulfillment Solution.

This chapter also provides the information required to complete [Step 13](#), [Step 14](#), and [Step 15](#) indicated in the [Installation Checklist](#).

If you need to deploy the Sterling Multi-Channel Fulfillment Solution in a development environment using exploded (non-ear) mode, see the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.

Before deployment, verify if you have applied all the concepts that pertain to your environment, and have completed the Performance Recommendations Checklist as described in the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

Tip: To enable faster loading of a JSP page, pre-compile your JSP files. For information on how to do this, see the JSP Pre-compilation section of the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

Note: If you are planning on installing any of the Sterling Multi-Channel Fulfillment Solution Packaged Composite Application (PCAs), or applying any extensions, you may want to consider delaying the building of your Enterprise Archive (EAR) until all of your PCAs are installed. Building the EAR now and for each PCA or extension installation does not cause harm, but does save time if you build your EAR only once after all PCAs or extensions are installed.

14.1 Setting Up the WebLogic Application Server

The following sections include information for setting up the WebLogic application server, including:

- “Setting Up the WebLogic Script File” on page 176
- “Configuring WebLogic for the Sterling Multi-Channel Fulfillment Solution” on page 178
- “Disabling Instrumented Stack Traces in WebLogic” on page 178
- “Setting Up WebLogic to Display Barcodes and Graphs” on page 179
- “Setting Up WebLogic to Use HTTP In-Memory Session Replication” on page 179

14.1.1 Setting Up the WebLogic Script File

If you are using HP-UX 11i v2, verify that your kernel parameters are set according to BEA's recommendations before you set up the WebLogic application server. For these recommendations, go to http://e-docs.bea.com/platform/suppconfigs/configs/hpux/hpux_11iv2_92.html.

To set up the WebLogic script file:

1. Add the following properties to the startWebLogic.sh (or .cmd) file supplied by BEA. Each property and its proper syntax are described in the following table.

Property	Description
JAVA_OPTIONS	<p>Java command line options for running the server.</p> <p>Depending on your JVM vendor, specify as follows:</p> <ul style="list-style-type: none"> • For IBM, set this value to <code>-Xms512m -Xmx512m</code> • For JRockit, set this value to <code>-Xms512m -Xmx512m</code> • For HP, set this value to <code>-XX:MaxPermSize=256m -Xms512m -Xmx512m</code> • For Sun, set this value to <code>-XX:MaxPermSize=256m -Xms512m -Xmx512m</code>
DBDRIVERS	<p>Specify the paths to your data base drivers as the first item in the value of the CLASSPATH.</p> <p>The out-of-the-box CLASSPATH setting is:</p> <pre>CLASSPATH="\${CLASSPATH}\${CLASSPATHSEP}\${MEDREC_WEBLOGI C_CLASSPATH}"</pre> <p>Change this so that the path to the drivers .jars is first. For example:</p> <pre>DBDRIVERS=/u01/home/jsmith/lib/oracle/ojdbc14.jar CLASSPATH="\${DBDRIVERS}\${CLASSPATHSEP}\${CLASSPATH}\${CL ASSPATHSEP}\${MEDREC_WEBLOGIC_CLASSPATH}"</pre>
JITC_COMPILEOPTS	For AIX, specify as "NQCLSINIT"
-Dfile.encoding	<p>To ensure that all the Sterling Multi-Channel Fulfillment Solution UI screens display UTF-8 characters, specify as follows for java commands:</p> <pre>-Dfile.encoding=UTF-8</pre> <p>This is applicable to all the Sterling Multi-Channel Fulfillment Solution Java start-up scripts.</p>
-Dvendor	<p>System property. Specify as an argument to the java command. Can be "shell" or "weblogic" depending upon whether datasource is being used or not.</p> <pre>-Dvendor=shell</pre>
-DvendorFile	<p>System property. Specify as an argument to the java command.</p> <pre>-DvendorFile=/servers.properties</pre>

Property	Description
-Dsci.naming.provider.url	Sets the local jndi that the server will register with. Specify as an argument to the java command using the name or ip of the server and the port it is listening on. -Dsci.naming.provider.url=t3://<hostname>:<port>

2. If you are using an HTTPS transport, download the Secure Socket Extension (JSSE) 1.0.3 package from <http://java.sun.com> and add the following files to the <JAVA_HOME>/jre/lib/extn/ directory:
 - jnet.jar
 - jcert.jar
 - jsse.jar

14.1.2 Configuring WebLogic for the Sterling Multi-Channel Fulfillment Solution

You must configure WebLogic to run properly with the Sterling Multi-Channel Fulfillment Solution.

To configure WebLogic:

1. From the WebLogic Console menu, choose Services > XML Registries.
2. Click New.

Note: You do *not* need to set an XML registry parameter for UTF-8. This is predefined.

3. Click Next. Select the WebLogic application server or cluster to which you would like to deploy this XML Registry.
4. Click Finish.

14.1.3 Disabling Instrumented Stack Traces in WebLogic

You can eliminate additional stack traces resulting from an error on an API call in EJB mode.

To eliminate stack traces:

1. From the WebLogic System Administration Console, select each server on which the Sterling Multi-Channel Fulfillment Solution is deployed.
2. Select Logging.
3. Uncheck the checkbox for Instrument Stack Traces and choose Apply.

14.1.4 Setting Up WebLogic to Display Barcodes and Graphs

The Sterling Multi-Channel Fulfillment Solution uses X Window functionality to display barcodes and dynamic graphical images (such as inventory supply and demand graphs) in a UNIX environment.

The following configuration is required to enable the X Window environment in UNIX systems for a WebLogic application server:

1. If your UNIX server is also an X Window client, edit the `startWebLogic.sh` script, and set the `DISPLAY` environment variable as follows:

```
export DISPLAY=IP_address_of_XWindows_server:0.0.
```

2. If you are using UNIX, run the `xhost +` command to remove access control for your X Window server.

You can run X server on the same server on which you run the Sterling Multi-Channel Fulfillment Solution. However, you need to be logged into the server console.

Note: If the X Window server goes down or crashes while the inventory user interface is using the `jbchartx.jar` file, the WebLogic server also goes down.

14.1.5 Setting Up WebLogic to Use HTTP In-Memory Session Replication

The Sterling Multi-Channel Fulfillment Solution supports HTTP in-memory session replication on the following configuration:

- Apache 2.0.44 with the WebLogic plug-in as the proxy server with `idempotent` set to OFF

We advise testing session replication if you are using a different proxy.

The `weblogic.xml` file should be edited to set up WebLogic for in-memory session replication as follows:

1. Copy the
`<INSTALL_DIR>/repository/eardata/yantra/descriptors/weblogic/WAR/WEB-INF/weblogic.xml` file to the
`<INSTALL_DIR>/extensions` directory.
2. Add the following lines to the `weblogic.xml` file:

```
<session-descriptor>
  <session-param>
    <param-name>PersistentStoreType</param-name>
    <param-value>replicated</param-value>
  </session-param>
</session-descriptor>
```
3. Rebuild the EAR file.

14.2 Building the Enterprise Archive (EAR) Package (WebLogic)

When deploying the Sterling Multi-Channel Fulfillment Solution on WebLogic, use the `yantra.ear` file, which contains:

- `yantra.war` - Web module that contains all of the Sterling Multi-Channel Fulfillment Solution JSPs and other Web application components.
- `yantraweb services.war` - Web module that contains all of the Sterling Multi-Channel Fulfillment Solution Web services interface classes.
- `yantraejb.jar` - The EJB module that contains all the Sterling Multi-Channel Fulfillment Solution EJBs.
- `yantrawsbe.jar` - The backend Web services jar file. You get this file if you expose Web services.
- Jars that contain backend business logic.
- Jars that contain third-party libraries accessed by backend logic.

Each of the third-party JAR files are left as is and in the manifest of the application each file is indicated as a dependency. For example, log4j files are represented separately as log4j-1.2.12.jar with a dependency in the application.

14.2.1 Deploying the Online Help (WebLogic)

The Sterling Multi-Channel Fulfillment Solution online help documentation is built as a separate EAR file called `yantradocs.ear` when you build the `yantra.ear` file unless you specify "`-Dnodecear=Y`" during the build. You can also build the doc ear separately by running the following command:

```
<INSTALL_DIR>/bin/buildear.sh (or cmd) create-doc-ear
-Dappserver=weblogic -Dwls-10=<true|don't pass if not wls10>
-Dpackage=yantra
```

To make use of the documentation and online help files associated with the Sterling Multi-Channel Fulfillment Solution, deploy the `yantradocs.ear` file in all of the same locations where you deploy the `yantra.ear` file.

14.2.2 Preparing to Build Web Services (WebLogic)

If you are planning to run the Sterling Multi-Channel Fulfillment Solution components as Web services, additional setup is required. The setup takes place in the `namedwebservices.xml` file.

You can expose either all or selected APIs as Web services. Consider the following:

- To expose selected APIs as Web services, set the `ExposeAllAPIs` attribute value to `N` and specify each API you want to expose in an `Api/Name` attribute.
- To expose all the Sterling Multi-Channel Fulfillment Solution APIs as Web services, set the `ExposeAllAPIs` attribute value to `Y`. If the `ExposeAllAPIs` attribute is set to `Y`, all `<Api>` node attributes are ignored.

If you are exposing individual services, edit the attributes of the `namedwebservices.xml` file, as described in the following table, before you create your `yantra.ear` file.

Property	Description
ServiceName	The name of the service that you configured using the Sterling Multi-Channel Fulfillment Solution Service Builder.
ExposedName	The name that is used in the Web Services Description Language (WSDL) file. This is the name that is used to call the webservice programmatically. When specifying a service name for ExposedName, choose a literal that does not match any of the standard Sterling Multi-Channel Fulfillment Solution API names.

To configure the Sterling Multi-Channel Fulfillment Solution as Web services:

1. Edit the `<INSTALL_DIR>/properties/sandbox.cfg` file to set the value of the `BEA_DIR` property to specify the `<WL_HOME>` directory and run the `<INSTALL_DIR>/bin/setupfiles.sh` (or `setupfiles.cmd`) script.
2. Rename the `<INSTALL_DIR>/repository/eardata/platform/webservices/namedwebservices.xml.sample` file to `namedwebservices.xml`.
3. Edit the `namedwebservices.xml` file to update the exposed names to start with a lowercase letter.

To have the Sterling Multi-Channel Fulfillment Solution run services as Web services on WebLogic:

1. Edit the `<INSTALL_DIR>/repository/eardata/platform/webservices/namedwebservices.xml` file to remove any Sterling Multi-Channel Fulfillment Solution APIs that you do not want exposed as named Web services and include the Services you want to expose as named Web services. All the Sterling Multi-Channel Fulfillment Solution APIs are available to run as Web services.
2. Create the EAR as described in [Section 14.2.4, "Creating the EAR \(WebLogic\)"](#).
3. Save the EAR file to a new location.
4. If you want to expose different APIs and services as webservices, repeat these steps as many times as desired.

14.2.3 Including Custom Classes (WebLogic)

When deploying the Sterling Multi-Channel Fulfillment Solution as Web services on WebLogic, if you have developed custom Java classes (user exits, event handlers, and so forth) you need to deploy them in order for them to be available.

To ensure that your custom classes get invoked, do the following:

1. Create a JAR file with all your custom classes.
2. Place this JAR file in a folder structure based on the package name. For more information about packaging and deploying jar files, see the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.
3. Rebuild the EAR as described in [Section 14.2.4, "Creating the EAR \(WebLogic\)"](#).

The custom classes are automatically included in the `yantra.ear` file.

14.2.4 Creating the EAR (WebLogic)

Note: Set the number of file descriptors (`ulimit -n`) for the user creating the EAR to be greater than 8192. If you are deploying on HP set `ulimt unlimited` for the user creating the EAR.

To create an application EAR file:

1. Run the following command from the `<INSTALL_DIR>/bin` to create the EAR file:

```
./buildear.sh (cmd for Windows) -Dappserver=weblogic
-Dwls-10=<true|don't pass if not wls10> -Dpackage=yantra
create-ear
```

Note: Add the option `-Dnowbservice=true` to the end of the command above if you do not want to use Web services. If you do want to use Web services, see the [Section 14.2.2, "Preparing to Build Web Services \(WebLogic\)"](#).

Running this command creates two EAR files called `yantra.ear` and `yantradocs.ear` in the `<INSTALL_DIR>/external_deployments/` directory.

For more information, access the WebLogic documentation at http://edocs.bea.com/wls/docs92/quickstart/quick_start.html or <http://edocs.bea.com/wls/docs100/index.html> (the "J2EE Deployment" section).

The following table shows the actions taken by this command through the `buildEAR.xml` file.

Main Target	Description
create-ear	Creates the EAR file (<code>yantra.ear</code>) for deployment.
create-doc-ear	Creates the Sterling Multi-Channel Fulfillment Solution documentation EAR file (<code>yantradocs.ear</code>).

14.2.4.1 Installing and Deploying Sterling Multi-Channel Fulfillment Solution on Different Servers

This section applies only to users who are installing and deploying the Sterling Multi-Channel Fulfillment Solution on separate systems.

In order for this scenario to deploy successfully, you must identify a log directory on the system where you are deploying Sterling Multi-Channel Fulfillment Solution.

1. Set the `LOG_DIR` property in `sandbox.cfg` to a value that is meaningful on the system where the EAR will be deployed.
2. Run the `<INSTALL_DIR>/bin/setupfiles.sh` (or `setupfiles.cmd`) script.
3. Rebuild the EAR file.
4. Edit `sandbox.cfg` to set the `LOG_DIR` value back to its original value.

For more information about `sandbox.cfg` and changing properties, see [Chapter 12, "Configuring Properties"](#) and for information about individual properties in `sandbox.cfg`, see [Appendix A, "Property Files Reference"](#).

14.2.5 Precompiling the WAR File (WebLogic)

To improve the performance when initially loading UI resources, Sterling Commerce recommends that you precompile the jsps that comprise the WAR file. For more information about how to pre-compile jsps, see *"JSP Pre-Compilation"* in the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

Deploy the new ear file as described in [Section 14.3, "Deploying the Enterprise Archive \(EAR\) \(WebLogic\)"](#).

14.3 Deploying the Enterprise Archive (EAR) (WebLogic)

For instructions on deploying the EAR to your WebLogic application server, see your WebLogic documentation.

To verify the Sterling Multi-Channel Fulfillment Solution installation:

1. Restart your application server.
2. Start Internet Explorer.
3. Access `http://<hostname>:<port>/yantra/console/login.jsp`.
4. When prompted, enter your Login ID and Password. If the Sterling Multi-Channel Fulfillment Solution Administrator's home page is not displayed, contact the Sterling Multi-Channel Fulfillment Solution Technical Support Services at:
`http://www.sterlingcommerce.com/scm_support/`

14.4 Setting Up the WebSphere Application Server

Before configuring WebSphere, Sterling Commerce recommends that you start the WebSphere administrative server with the following memory parameters:

- -Xms512 MB or higher
- -Xmx512 MB or higher

You have the option to avoid the warning messages regarding direct datasource lookups that occur at run time. To avoid these messages, do the following:

1. From the WebSphere Administrative Console, expand Troubleshooting in the left panel and click on Logs and Trace.
2. Select each server that hosts the Sterling Multi-Channel Fulfillment Solution and choose Change Log Detail Levels in the General Properties.
3. In the Components panel, select the class, `com.ibm.ejs.j2c.ConnectionFactoryBuilderImpl`, and specify the log level as severe.
4. Save the changes to the Master Configuration.

Also ensure that the WebSphere Classloader is set correctly for Classloader policy and Class loading modes as follows:

1. From the Administrative Console left panel, choose Servers => Application Servers.
2. Select among the servers listed.
3. Set the Classloader policy pulldown to Single and the Class loading mode pulldown to Parent first.

The following sections include additional information for setting up the WebSphere application server, including:

- ["Application Clients Invoking the Sterling Multi-Channel Fulfillment Solution EJBs" on page 186](#)
- ["Configuring WebSphere JVM Settings" on page 187](#)
- ["Configuring WebSphere for Running the Sterling Multi-Channel Fulfillment Solution Agents" on page 188](#)
- ["Setting Up WebSphere to Display Barcodes and Graphs" on page 189](#)
- ["Configuring DataSource Connection Pooling on WebSphere" on page 190](#)

14.4.1 Application Clients Invoking the Sterling Multi-Channel Fulfillment Solution EJBs

In order to make EJB calls in the Sterling Multi-Channel Fulfillment Solution using WebSphere you need to generate EJB stubs and skeletons. The following steps outline the method for creating the JAR files using the `ejbdeploy.sh` script to generate the stubs:

1. Set the CLASSPATH to include xercesImpl.jar, xalan.jar, and xml-apis.jar as provided in the *JRE/lib/endorsed* directory. Also, CLASSPATH must include the jar files specified in the dynamicclasspath.cfg file.
2. Invoke the ejbdeploy.sh command from the <WebSphere_Home>/bin directory with the following three arguments:
 - a. Specify the full path to the yantraejb.jar file in <INSTALL_DIR>/external_deployments/ directory.
 - b. Specify the temporary directory that is used for the EJB deployment.
 - c. Specify the full path to the desired output file, for example yantra_ejbstubs.jar.

Additionally set the classpath on the ejbdeploy.sh command line following the -cp argument. For example:

```
$WebSphere_Home/bin/ejbdeploy.sh
<INSTALL_DIR>/external_deployments/yantraejb.jar WebSphere_Home/temp
<INSTALL_DIR>/external_deployments/yantraejb.jar -cp $CLASSPATH
```

14.4.2 Configuring WebSphere JVM Settings

You need to use the WebSphere Administrative Console to specify the JVM settings. These JVM settings must be set on **all** servers in a cluster (if you are using a cluster).

To configure JVM setting on WebSphere, do the following:

1. From the WebSphere Administrative Console, select the application server specified for the Sterling Multi-Channel Fulfillment Solution.
2. For IBM servers with IBM JDK 1.5:
 - a. Select Server Infrastructure > Java and Process Management > Process Definition > Environment Entries.
 - b. Choose **New** and specify the following values and then choose **OK**:

Name	Value	Description
PSALLOC	early	PSALLOC
NODISCLAIM	true	NODISCLAIM

3. Select Server Infrastructure > Java and Process Management > Process Definition > Java Virtual Machine. Edit the generic JVM arguments dialog to include the following values:

Property	Value	Description
-Dvendor	-Dvendor=shell	System Property. If you are using App Server Connection Pooling, use -Dvendor=websphere. Otherwise, use -Dvendor=shell.
-DvendorFile	-DvendorFile=/servers.properties	System property. Specify as an argument to the java command.
-Dsci.naming.provider.url	-Dsci.naming.provider.url=corbaloc::host:port	Sets the local jndi that the server will register with. Specify as an argument to the java command using the name or IP of the server and the port it is listening on.

4. Under the Custom Properties section, set the JVM settings to the following values:

Name	Value	Description
client.encoding.override	UTF-8	Enables the use of special characters.

5. Restart the application server to enable these changes to take effect.
6. Save the changes to the Master Configuration.

14.4.3 Configuring WebSphere for Running the Sterling Multi-Channel Fulfillment Solution Agents

The Sterling Multi-Channel Fulfillment Solution requires both WebSphere and MQ jars for running the agent and integration servers.

14.4.4 Setting Up WebSphere to Display Barcodes and Graphs

The Sterling Multi-Channel Fulfillment Solution uses the X Window functionality to display barcodes and dynamic graphical images (such as inventory supply & demand graphs) in a UNIX environment.

The following configuration is required to enable the X Window environment in UNIX systems for the WebSphere application server:

1. From the WebSphere Administrative Console, go to Servers > Application Server and select the application server specified for the Sterling Multi-Channel Fulfillment Solution.
2. On the Configuration tab, select Java and Process Management under Server Infrastructure option.
3. Select Process Definition.
4. On the configuration, go to Additional Properties and select Environment Entries.
5. Select New.
6. On the General Properties enter the Name as DISPLAY and the value as *IP_address_of_XWindows_server*:0.0. Do make sure that the X Window server accepts requests from this client.
7. If you are using UNIX, run the `xhost+` command to remove access control for your X Window server.

You can run X server on the same server in which you run the Sterling Multi-Channel Fulfillment Solution. However, you need to be logged to the server console.

Restart the application server for the DISPLAY variable to take effect.

8. Save the changes to the Master Configuration.

Note: If the X Window server goes down or crashes while the inventory user interface is using the `jbchartx.jar` file, the WebSphere server also goes down.

14.4.5 Configuring DataSource Connection Pooling on WebSphere

If you are using DataSource Connection Pooling, perform the following steps:

1. In the `customer_overrides.properties` file, add the following lines:
For DB2, add:
 - `jdbcService.db2Pool.datasource=<datasourceName>`For Oracle, add:
 - `jdbcService.OraclePool.datasource=<datasourceName>`For detailed information about using the `customer_overrides.properties` file, see [Chapter 12, "Configuring Properties"](#).
2. Configure the DataSource on the WebSphere Application Server. The JNDI Name must be `datasourcename`.
3. Test the connection.
4. To avoid the warning messages regarding direct datasource lookups that occurs at run time:
 - a. From the WebSphere Administrative Console, expand Troubleshooting in the left panel and click on Logs and Trace.
 - b. Select each server that hosts the Sterling Multi-Channel Fulfillment Solution and choose Change Log Detail Levels in the General Properties.
 - c. In the Components panel, select the class, `com.ibm.ejs.j2c.ConnectionFactoryBuilderImpl`, and specify the log level as severe.
5. Save the changes to the Master Configuration.

14.5 Building the Enterprise Archive (EAR) Package (WebSphere)

When deploying the Sterling Multi-Channel Fulfillment Solution on WebSphere, use the `yantra.ear` file, which contains:

- `yantra.war` - Web module that contains all of the Sterling Multi-Channel Fulfillment Solution JSPs and other Web application components.
- `yantrawebsservices.war` - Web module that contains all of the Sterling Multi-Channel Fulfillment Solution Web services interface classes.
- `yantraejb.jar` - The EJB module that contains all the Sterling Multi-Channel Fulfillment Solution EJBs.
- `yantrawsbe.jar` - The backend Web services jar file. You get this file if you expose Web services.
- jars that contain backend business logic
- jars that contain third-party libraries accessed by backend logic

Each of the third-party JAR files is left as is and in the manifest of the application each file is indicated as a dependency. For example, `log4j` files are represented separately as `log4j-1.2.12.jar` with a dependency in the application.

14.5.1 Deploying the Online Help (WebSphere)

The Sterling Multi-Channel Fulfillment Solution online help documentation is built as a separate EAR file called `yantradocs.ear` using the command:

```
<INSTALL_DIR>/bin/buildear.sh -Dappserver=websphere
-Dpackage=yantra create-doc-ear
```

To make use of the documentation and online help files associated with the Sterling Multi-Channel Fulfillment Solution, deploy the `yantradocs.ear` file in all of the same locations where you deploy the `yantra.ear` file.

14.5.2 Preparing to Build Web Services (WebSphere)

If you are planning to run the Sterling Multi-Channel Fulfillment Solution components as Web services, additional setup is required. The setup takes place in the `namedwebservices.xml` file.

You can expose either all or selected APIs as Web services. Consider the following:

- To expose selected APIs as Web services, set the `ExposeAllAPIs` attribute value to `N` and specify each API you want to expose in an `Api/Name` attribute.
- To expose all the Sterling Multi-Channel Fulfillment Solution APIs as Web services, you can set the `ExposeAllAPIs` attribute value to `Y`. If the `ExposeAllAPIs` attribute is set to `Y`, all `<Api>` node attributes are ignored.

If you are exposing individual services, edit the attributes of the `namedwebservices.xml` file, as described in the following table, before you create your `yantra.ear` file.

Property	Description
ServiceName	The name of the service that you configured using the Sterling Multi-Channel Fulfillment Solution Service Builder.
ExposedName	The name that is used in the Web Services Description Language (WSDL) file. This is the name that is used to call the web service programmatically. When specifying a service name for <code>ExposedName</code> , choose a literal that does not match any of the standard Sterling Multi-Channel Fulfillment Solution API names.

To configure the Sterling Multi-Channel Fulfillment Solution as Web services:

1. Edit the `<INSTALL_DIR>/properties/sandbox.cfg` file to set the value of the `WAS_DIR` property to specify the `<WAS_HOME>` directory and run the `<INSTALL_DIR>/bin/setupfiles.sh` (or `setupfiles.cmd`) script.
2. Rename the `<INSTALL_DIR>/repository/eardata/platform/webservices/namedwebservices.xml.sample` file to `namedwebservices.xml`.

3. Edit the `namedwebservices.xml` file to update the exposed names to start with a lowercase letter.

14.5.3 Including Custom Classes (WebSphere)

If you have developed custom Java classes (user exits, event handlers, and so forth) you need to deploy them in order for them to be available.

To ensure that your custom classes get invoked, do the following:

1. Create a JAR file with all your custom classes.
2. Use `install3rdParty.sh` to include your custom jar file in the APP classpath. For more information about packaging and deploying jar files, see the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.
3. Rebuild the EAR as described in [Section 14.5.4, "Creating the EAR \(WebSphere\)"](#).

The custom classes are automatically included in the `yantra.ear` file.

14.5.4 Creating the EAR (WebSphere)

Enterprise Archives are built using an ANT xml that accepts the following targets:

Main Target	Description
create-ear	Creates the EAR file (<code>yantra.ear</code>) for deployment.
create-doc-ear	Creates the Sterling Multi-Channel Fulfillment Solution documentation EAR (<code>yantradocs.ear</code>).

To create an application EAR file:

1. Run the following command from the `<INSTALL_DIR>/bin` to create the EAR file:

```
./buildear.sh -Dappserver=websphere -Dpackage=yantra
create-ear
```

Note: Add the option `-Dnowebservice=true` to the end of the command above if you do not want to use Web Services. If you do want to use Web services, see [Section 14.5.2, "Preparing to Build Web Services \(WebSphere\)"](#) on page 192.

Note: Add the option `-Dnodocear=true` to the end of the above command if you want to skip the documentation build.

Running this command creates the EAR file called `yantra.ear` in the `<INSTALL_DIR>/external_deployments/` directory.

14.5.4.1 Installing and Deploying Sterling Multi-Channel Fulfillment Solution on Different Servers

This section applies only to users who are installing and deploying the Sterling Multi-Channel Fulfillment Solution on separate systems.

In order for this scenario to deploy successfully, you must identify a log directory on the system where you are deploying Sterling Multi-Channel Fulfillment Solution.

1. Set the `LOG_DIR` property in `sandbox.cfg` to a value that is meaningful on the system where the EAR will be deployed.
2. Run the `<INSTALL_DIR>/bin/setupfiles.sh` script.
3. Rebuild the EAR file.
4. Edit `sandbox.cfg` to set the `LOG_DIR` value back to its original value.

For more information about `sandbox.cfg` and changing properties, see [Chapter 12, "Configuring Properties"](#) and for information about individual properties in `sandbox.cfg`, see [Appendix A, "Property Files Reference"](#).

14.5.5 Precompiling the WAR File (WebSphere)

To improve the performance when initially loading UI resources, Sterling Commerce recommends that you precompile the jsp's that comprise the WAR file. For more information about how to pre-compile jsp's, see *"JSP Pre-Compilation"* in the *Sterling Multi-Channel Fulfillment Solution Performance Management Guide*.

14.6 Deploying the Enterprise Archive (EAR) (using the WebSphere Admin Console)

To deploy the EAR on WebSphere:

1. From the WebSphere Administrative Console menu in the left pane, select Applications > Install New Application.
2. The right pane is populated with the specifics for the EAR location.
3. Choose Local File System or Remote File System. Click the corresponding Browse button and browse to the enterprise archive such as `yantra.ear` you want to deploy. Click Next.
4. Check Deploy enterprise beans, and if desired, change the application name. If you are using Web services, check Deploy WebServices and click Next.
5. The Map Modules to Servers screen displays. Select the checkbox next to each desired module (at least two entries, `yantraejb.jar` and `yantra.war`, should be present). Click the Cluster/Server in the Cluster and Server pane. Click Apply. The screen refreshes and the server field is updated with the chosen value. Click Next.
6. Accept the default JNDI names for the EJB modules on the Provide JNDI Names for Beans screen. Click Next.
7. On the Map Virtual Hosts for Web Modules screen, select your web module and its correct virtual host. Choose Next.
8. The Ensure all Unprotected 2.x Methods screen displays. Click Next.
9. The Provide Options to perform the WebServices Deployment screen displays. Leave them as is and click Next.
10. On the summary page, choose Finish.

To verify the Sterling Multi-Channel Fulfillment Solution installation:

1. Restart your application server.
2. Start Internet Explorer.
3. Access `http://<hostname>:<port>/yantra/console/login.jsp`.
4. When prompted, enter your Login ID and Password. If the Sterling Multi-Channel Fulfillment Solution Administrator home page is not displayed, contact the Sterling Multi-Channel Fulfillment Solution Technical Support Services at:
`http://www.sterlingcommerce.com/scm_support/`

14.7 Setting Up the JBoss Application Server

Note: The JBoss server must have the default name of "all" for the precompilation scripts to run successfully.

To set up the JBoss application server, you must set up some properties in the JBoss script file.

To set up the JBoss script file, do the following:

1. Add the following properties to the <JBOSS_HOME>/bin/run.sh (or .cmd) file supplied by JBoss. Each property and its proper syntax are described in the following table:

Property	Required Edits
JAVA_OPTS	Based on your JVM vendor, specify: IBM: <code>-Xms512m -Xmx512m</code> Sun: <code>-XX:MaxPermSize=256m -Xms512m -Xmx512m</code>
-Dfile.encoding	To ensure that all the Sterling Multi-Channel Fulfillment Solution screens display UTF-8 characters for java commands, specify: <code>-Dfile.encoding=UTF-8</code> This is applicable to all the Sterling Multi-Channel Fulfillment Solution Java start-up scripts.
-Dvendor	System Property. If you are using App Server Connection Pooling, use <code>-Dvendor=jboss</code> . Otherwise, use <code>-Dvendor=shell</code> . <code>-Dvendor=shell</code>
-DvendorFile	System property. Specify as an argument to the java command. <code>-DvendorFile=/servers.properties</code>
-Dsci.naming.provider.url	Sets the local jndi that the server will register with. Specify as an argument to the java command using the name or IP of the server and the port it is listening on. <code>-Dsci.naming.provider.url=jnp://host:portnum</code>

2. If you are using an HTTPS transport, download the Secure Socket Extension (JSSE) 1.0.3 package from <http://java.sun.com> and add the following files to the <JAVA_HOME>/jre/lib/extn/ directory:

- jnet.jar
- jcert.jar
- jsse.jar

In addition to this setup, see [“Setting Up JBoss to Display Barcodes and Graphs”](#) on page 197 for information about setting up the JBoss application server to display barcodes and graphs.

14.7.1 Setting Up JBoss to Display Barcodes and Graphs

The Sterling Multi-Channel Fulfillment Solution uses the X Window functionality to display barcodes and dynamic graphical images (such as inventory supply & demand graphs) in a UNIX environment. The following configuration is required to enable the X Window environment in UNIX systems for JBoss application servers:

1. If your UNIX server is also an X Window client, edit the `run.sh` script, and set the `DISPLAY` environment variable as follows:

```
export DISPLAY=<IP_address_of_XWindows_server>:0.0
```
2. If you are using UNIX, run the `xhost +` command to remove access control for your X Window server.
3. You can run X-server on the same server in which you run the Sterling Multi-Channel Fulfillment Solution. However, you need to be logged in to the server console.

Note: If the X Window server goes down or crashes while the inventory user interface is using the `jbchartx.jar` file, the JBoss server also goes down.

14.8 Building the Enterprise Archive (EAR) Package (JBoss)

When deploying the Sterling Multi-Channel Fulfillment Solution on JBoss, use the `yantra.ear` file, which contains:

- `sscapshared.jar` - Application module that contains all of the Sterling Multi-Channel Fulfillment Solution Java classes that encapsulate the core business logic.

- `yantra.war` - Web module that contains all of the Sterling Multi-Channel Fulfillment Solution JSPs and other Web application components.
- `yantraweb services.war` - Web module that contains all of the Sterling Multi-Channel Fulfillment Solution Web services interface classes.
- `yantraejb.jar` - The EJB module that contains all the Sterling Multi-Channel Fulfillment Solution EJBs.
- `yantrawsbe.jar` - The backend Web services jar file. You get this file if you expose Web services.
- Jars that contain backend business logic.
- Jars that contain third-party libraries accessed by backend logic.

Each of the third-party JAR files are left as is and in the manifest of the application each file is indicated as a dependency. For example, `log4j` files are represented separately as `log4j-1.2.12.jar` with a dependency in the application.

14.8.1 Deploying the Online Help (JBoss)

The Sterling Multi-Channel Fulfillment Solution online help documentation is built as a separate EAR file called `yantradocs.ear` using the command:

```
<INSTALL_DIR>/bin/buildear.sh create-doc-ear -Dappserver=jboss  
-Dpackage=yantra
```

To make use of the documentation and online help files associated with the Sterling Multi-Channel Fulfillment Solution, deploy the `yantradocs.ear` file in all of the same locations where you deploy the `yantra.ear` file.

14.8.2 Preparing to Build Web Services (JBoss)

If you are planning to run the Sterling Multi-Channel Fulfillment Solution components as Web services, additional setup is required. The setup takes place in the `namedwebservices.xml` file.

You can expose either all or selected APIs as Web services. Consider the following:

- To expose selected APIs as Web services, set the `ExposeAllAPIs` attribute value to `N` and specify each API you want to expose in an `Api/Name` attribute.
- To expose all the Sterling Multi-Channel Fulfillment Solution APIs as Web services, you can set the `ExposeAllAPIs` attribute value to `Y`. If the `ExposeAllAPIs` attribute is set to `Y`, all `<Api>` node attributes are ignored.

If you are exposing individual services, edit the attributes of the `namedwebservices.xml` file, as described in the following table, before you create your `yantra.ear` file.

Property	Description
ServiceName	The name of the service that you configured using the Sterling Multi-Channel Fulfillment Solution Service Builder.
ExposedName	The name that is used in the Web Services Description Language (WSDL) file. This is the name that is used to call the web service programmatically. When specifying a service name for <code>ExposedName</code> , choose a literal that does not match any of the standard Sterling Multi-Channel Fulfillment Solution API names.

To configure the Sterling Multi-Channel Fulfillment Solution as Web services:

1. Edit the `<INSTALL_DIR>/properties/sandbox.cfg` file to set the value of the `JBOSS_DIR` property to specify the `<JBOSS_HOME>` directory and run the `<INSTALL_DIR>/bin/setupfiles.sh` (or `setupfiles.cmd`) script.
2. Rename the `<INSTALL_DIR>/repository/eardata/platform/webservices/namedwebservices.xml.sample` file to `namedwebservices.xml`.
3. Edit the `namedwebservices.xml` file to update the exposed names to start with a lowercase letter.

To have the Sterling Multi-Channel Fulfillment Solution run services as Web services on JBoss:

1. Edit the `<INSTALL_DIR>/repository/eardata/platform/webservices/namedwebservices.xml` file to remove any Sterling Multi-Channel Fulfillment

Solution APIs that you do not want exposed as named Web services and include the Services you want to expose as named Web services. All the Sterling Multi-Channel Fulfillment Solution APIs are available to run as Web services.

2. Create the EAR as described in [Section 14.8.4, "Creating the EAR \(JBoss\)"](#).
3. Save the EAR file to a new location.
4. If you want to expose different APIs and services as webservices, repeat these steps as many times as desired.

14.8.3 Including Custom Classes (JBoss)

When deploying the Sterling Multi-Channel Fulfillment Solution as Web service on JBoss, if you have developed custom Java classes (user exits, event handlers, and so forth) you need to deploy them in order for them to be available.

To ensure that your custom classes get invoked, do the following:

1. Create a JAR file with all your custom classes.
2. Place this JAR file in a folder structure based on the package name. For more information about packaging and deploying jar files, see the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.
3. Rebuild the EAR as described in [Section 14.8.4, "Creating the EAR \(JBoss\)"](#).

These classes are automatically included in the `yantra.ear` file.

14.8.4 Creating the EAR (JBoss)

To create a `yantra.ear` file on JBoss:

- If you are building webservices:
 1. Add the following three variables to the `<INSTALL_DIR>/properties/sandbox.cfg` file:
 - `JBoss_DIR` - set to the absolute path of the JBoss installation directory.

- JBOSS_PRECOMPILE_JSP - set to true (precompile jsps) or false (do not precompile jsps). There is no default set by installation, but jsps will not be precompiled unless you set this to true.
- EJB_3_ENABLED - set to true or false. The flag determines if the ejbs are generated according to the spec version 2 or 3. Jboss supports both.

See [Chapter 12, "Configuring Properties"](#) for more information about editing the sandbox.cfg file.

2. Run the `<INSTALL_DIR>/bin/setupfiles.sh` script.
3. Run the following command from the `<INSTALL_DIR>/bin` to create the EAR file:

```
/buildear.sh -Dappserver=jboss -Dpackage=yantra create-ear
```

Note: Add the option `-Dnowebservice=true` to the end of the command above if you do not want to use Web Services. If you do want to use Web services, see [Section 14.8.2, "Preparing to Build Web Services \(JBoss\)"](#) on page 197.

Running this command creates the EAR file called `yantra.ear` in the `<INSTALL_DIR>/external_deployments/` directory.

- If you have extended your database:
1. Run the following command from the `<INSTALL_DIR>/bin` directory:

```
buildear.sh -Dpackage=yantra -Dappserver=jboss -Dnowebservice=true  
create-ear
```

This creates a database extension JAR file in the `<INSTALL_DIR>/repository` directory.

2. Rebuild the EAR file.

The following table shows the actions taken by this command through the `buildEAR.xml` file.

Main Target	Description
create-ear	Creates the EAR file (<code>yantra.ear</code>) for deployment.

Main Target	Description
create-doc-ear	Creates the Sterling Multi-Channel Fulfillment Solution documentation EAR (<code>yantradocs.ear</code>).

14.8.4.1 Installing and Deploying Sterling Multi-Channel Fulfillment Solution on Different Servers

This section applies only to users who are installing and deploying the Sterling Multi-Channel Fulfillment Solution on separate systems.

In order for this scenario to deploy successfully, you must identify a log directory on the system where you are deploying Sterling Multi-Channel Fulfillment Solution.

1. Set the LOG_DIR property in `sandbox.cfg` to a value that is meaningful on the system where the EAR will be deployed.
2. Run the `<INSTALL_DIR>/bin/setupfiles.sh` script.
3. Rebuild the EAR file.
4. Edit `sandbox.cfg` to set the LOG_DIR value back to its original value.

For more information about `sandbox.cfg` and changing properties, see [Chapter 12, "Configuring Properties"](#) and for information about individual properties in `sandbox.cfg`, see [Appendix A, "Property Files Reference"](#).

14.8.5 Precompiling the WAR File (JBoss)

See the *Performance Management Guide* for instructions on how to precompile the WAR on JBoss. There are settings that must be configured before you create the EAR file.

14.9 Deploying the Enterprise Archive (EAR) (JBoss)

Deploy your newly created `yantra.ear` file as described in your application server documentation, using the `deployEARJBoss.xml` ant script.

1. Stop the application server.
2. Copy the ear file to the deployment directory on the application server (the JBoss installation directory is `<servername>/deploy`).

3. Restart the application server.
4. Log in.

To verify the Sterling Multi-Channel Fulfillment Solution installation:

1. Restart your application server.
2. Start Internet Explorer.
3. Access `http://<hostname>:<port>/yantra/console/login.jsp`.
4. When prompted, enter your Login ID and Password. If the Sterling Multi-Channel Fulfillment Solution Administrator's home page is not displayed, contact the Sterling Multi-Channel Fulfillment Solution Technical Support Services at:
`http://www.sterlingcommerce.com/scm_support/`

14.10 Setting the Client Character Display

When displaying special characters, such as for various languages, the client computer must be configured to display these characters.

In order for Unicode characters to display correctly in the Sterling Multi-Channel Fulfillment Solution Consoles, each Windows client must be configured. To configure a client machine select Control Panel > Regional and Language Options.

You can use apostrophes and international characters in database queries, the Condition Builder, the Configuration Deployment Tool and other user interface fields.

14.11 Clearing Browser and Java Plugin Caches

Once the Sterling Multi-Channel Fulfillment Solution is ready for deployment, each user must clear the browser and Java Plugin caches on their client machines before launching the Sterling Multi-Channel Fulfillment Solution.

To clear the browser cache:

1. From the Windows start menu, select Settings > Control Panel > Internet Options. Choose the General tab, and in the Temporary Internet Files inner panel, choose the Delete Files button. The Delete Files dialog displays.

2. Enable the Delete All Offline Content option. Then click OK, and click OK once more.
3. Close the Internet Properties window.

To clear the Java plugin cache:

1. From the Windows start menu, select Settings > Control Panel > Java Plugin and choose the Cache tab.
2. Click Clear JAR Cache.
3. Click OK.
4. Close the Java Plugin Control Panel window.

14.12 Statistics Monitoring

In order to measure throughput performance, runtime statistics can be gathered. Note that this feature and the data gathered by it in the YFS_STATISTICS_DETAILS table are only for use by Sterling Commerce personnel, as any metric can change without notice.

In a production environment, you should leave statistics generation enabled to collect statistics data in 10 minute intervals (the default). You should also schedule statistic purges on a regular basis (for example, every two weeks).

Deploying and Updating the Sterling Rich Client Applications

This chapter explains how to deploy and update the Sterling Rich Client applications such as Sterling COM PCA (Packaged Composite Application) in different geographical locations.

This chapter also provides the information required to complete [Step 16](#) indicated on the [Installation Checklist](#) on page 1.

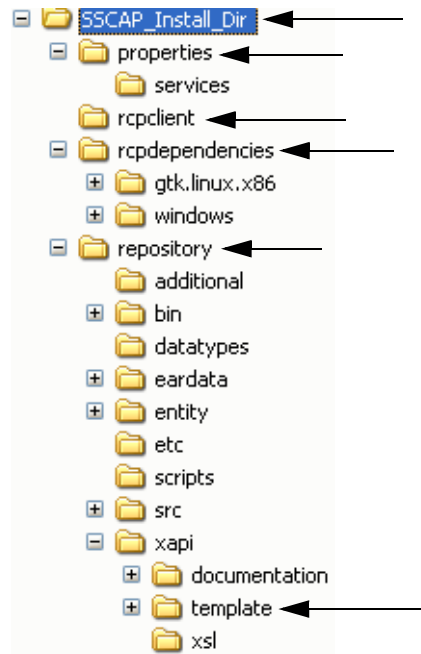
15.1 Before You Begin

Before you start deploying a Sterling Rich Client application you must have installed the Sterling Multi-Channel Fulfillment Solution. For more information about installing the Sterling Multi-Channel Fulfillment Solution, see [Chapter 7, "Installing in UNIX or Linux Environments"](#) or [Chapter 8, "Installing in a Windows Environment"](#).

The components that are shipped as part of foundation or platform (that is, what is available when the Sterling Multi-Channel Fulfillment Solution is installed) are:

- RCP Infrastructure plug-in's zip file
- RCP Foundation plug-in's zip file
- JREs for each of the supported operating systems. For more information about supported JREs and operating systems, see [Chapter 2, "System Requirements"](#).
- Eclipse dependencies for each of the supported operating systems

After you install the Sterling Multi-Channel Fulfillment Solution, you can view the directory structure as shown:



The directory structure contains:

- The `<INSTALL_DIR>` folder—This contains the Sterling RCP (Rich Client Platform) files, plug-ins, or JREs.
- The `<INSTALL_DIR>/rcpclient` folder—This contains the Sterling RCP plug-in and tools plug-in zip files.
- The `<INSTALL_DIR>/rcpdependencies` folder—This contains the Sterling RCP dependency directories for the supported operating systems. For example, `linux-gtk`, `windows`, and so forth. Each of these directories contains the supported JREs and Eclipse plug-ins, features, or files. Also, each of these directories contain the `osversion.properties` text file which provides information about the supported versions of the operating system.
- The `<INSTALL_DIR>/repository/xapi/template/merged/api` folder—This contains the API XML templates used by the Sterling RCP.
- The `<INSTALL_DIR>/properties` folder—This contains the `customer_overrides.properties` properties file. This file is used

when enabling auto updates for the individual PCA. For more information about enabling auto updates, see [Section 15.2.5, "Applying Updates"](#).

What is available when a Sterling RCP-based PCA is installed?

When you install a Sterling RCP-based PCA client, a zip file that contains the Sterling Rich Client application plug-ins or features is provided. For example, when you install the Sterling COM PCA application, the `<INSTALL_DIR>/rcp/COM/rcpclient` directory is automatically created. The `com.zip` file is stored in this directory, which contains the Sterling COM PCA-specific plug-ins or features.

15.2 Deploying the Sterling Rich Client Application

Deploying the Sterling Rich Client application involves:

- [Creating the RCP_EXTN_FOLDER Folder](#)
- [Configuring Locations](#)
- [Localizing Bundle and Theme Files](#)
- [Enabling HTTPS](#)
- [Applying Updates](#)
- [Running the Ant Script](#)

15.2.1 Creating the RCP_EXTN_FOLDER Folder

To maintain all SSL certificates, new plug-ins and new resource files that you created while extending Sterling RCP-based PCA client application, you must create a `<RCP_EXTN_FOLDER>` folder.

`RCP_EXTN_FOLDER` is the environment variable, which refers to the folder that you created to store files of the extended client application.

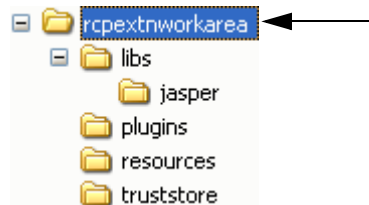
The `<RCP_EXTN_FOLDER>` folder structure is better explained with an example as follows:

1. Create an appropriate `<RCP_EXTN_FOLDER>` folder for storing the RCP extensions that you create when extending the Sterling RCP-based PCA client application. For example, `rcpextnworkarea` folder.

Note: You can create the <RCP_EXTN_FOLDER> folder in any directory outside the <INSTALL_DIR> directory.

2. Under the <RCP_EXTN_FOLDER> folder, create the following directories as illustrated in [Figure 15–1, "Sample <RCP_EXTN_FOLDER> Folder Structure"](#):
 - plugins
 - resources
 - truststore
 - libs

Figure 15–1 Sample <RCP_EXTN_FOLDER> Folder Structure



3. In the <RCP_EXTN_FOLDER>/plugins directory, store all new plug-ins that you created for extending the screens.
4. In the <RCP_EXTN_FOLDER>/resources directory, store the locations.ycfg file, secureapis.xml file (if necessary), localized bundle and theme files, and localized icons. The ant script creates the resources.jar file and copies the contents of the resources folder into this jar file. After copying the contents, the resources.jar file is copied into the Sterling RCP plug-in.
5. In the <RCP_EXTN_FOLDER>/truststore directory, store the SSL trust certificates that needs to be used when the client application is communicating with the server in secure mode. The SSL certificates are automatically copied by the ant script to the correct folder in the Sterling RCP plug-in.
6. Create the jasper folder within the <RCP_EXTN_FOLDER>/libs directory.

7. Copy the following jasper libs needed for JasperReports to the `<RCP_EXTN_FOLDER>/libs/jasper` folder:

- barbecue-1.1.jar
- commons-beanutils-1.5.jar
- commons-collections-2.1.jar
- commons-digester-1.7.jar
- commons-logging-1.0.2.jar
- iReport.jar
- itext-1.3.1.jar
- jasperreports-1.2.0.jar

To download these jasper libs, see the

`<INSTALL_DIR>/xapidocs/code_examples/jasperreports/readme.html` file.

15.2.1.1 Caching Data Types Locally in the Sterling RCP Based-PCA Client

To improve the system performance when logging into the Sterling RCP-based PCA application, you must cache data locally in the client.

To cache data types locally:

1. Copy the `datatypes.xml` file from the `<INSTALL_DIR>/repository/datatypes` folder, and `yfsdatatypepemap.xml` files from the `<INSTALL_DIR>/repository/xapi/template/merged/resource` directory to the `<RCP_EXTN_FOLDER>/resources` directory.
2. Create the `extn` directory under the `<RCP_EXTN_FOLDER>/resources` directory.
3. Copy the extended `datatypes.xml` and `yfsdatatypepemap.xml` from the `extensions` directory to the `<RCP_EXTN_FOLDER>/resources/extn` directory.

15.2.1.2 Supported Browser Version on Red Hat Linux Workstation 4

The default Firefox browser that is installed with the Red Hat Linux Workstation 4 is the certified version.

Note: You need to set the environment variable, MOZILLA_FIVE_HOME to the folder containing your Firefox installation. For example, set the env MOZILLA_FIVE_HOME as /usr/lib/firefox-1.5.0.3.

For more information about the supported browser version, see the following links:

- <http://www.eclipse.org/swt/faq.php#browserlinux>
- <http://www.eclipse.org/swt/faq.php#browserlinuxrcp>

15.2.1.3 Supported Browser Version on Suse Linux Desktop 10

Sterling RCP supports Mozilla browser on the Suse Linux Desktop 10. The default Firefox browser that is installed with the Suse Linux Desktop 10 is not supported.

Note: You need to set the environment variable, MOZILLA_FIVE_HOME to the folder containing your Mozilla installation. For example, set the env MOZILLA_FIVE_HOME as /usr/lib/mozilla-1.7.12.

For more information about the supported browser version, see the following links:

- <http://www.eclipse.org/swt/faq.php#browserlinux>
- <http://www.eclipse.org/swt/faq.php#browserlinuxrcp>

15.2.1.4 Installing Browser Plugins on Linux

In the Sterling RCP-based applications, FusionCharts, and JasperReports can be viewed in the client application. This requires the following plugins to be installed on the Firefox browser:

- Flash Player Plugin to view FusionCharts
- Mozplugger to view JasperReports

15.2.1.5 Installing the Flash Player Plugin

To install the flash player plugin, open the following link in your Linux Firefox browser and follow the instructions provided:

http://www.adobe.com/shockwave/download/download.cgi?P1_Prod_Version=ShockwaveFlash

Note: During installation, the browser installation directory should point to the Firefox install directory. For example, `/usr/lib/firefox-1.5.0.3`.

15.2.1.6 Installing Mozplugger

To install the Mozplugger:

1. Navigate to: <http://mozplugger.mozdev.org/>
2. Download Version 1.7.3 SOURCE of the MozPlugger to a local directory. The `mozplugger-1.7.3.tar.gz` file is downloaded.
3. Untar the `mozplugger-1.7.3.tar.gz` file. The `mozplugger-1.7.3` directory is created.
4. Change the directory to `mozplugger-1.7.3` and run the following commands:
 - `make linux`
 - `make install`

Note: During installation, the browser installation directory should point to the Firefox install directory. For example, `/usr/lib/firefox-1.5.0.3`.

15.2.2 Configuring Locations

A location is synonymous to a geographic location. For example, store location, call center location, and so forth. Each location has an identifier associated with it, which uniquely identifies the appropriate geographical location.

To configure locations, you must define locations in the `locations.ycfg` file. By default, the `locations.ycfg.sample` file is shipped by the Sterling RCP. You can locate this file in the Sterling RCP plug-in directory. The file in which the `locations.ycfg.sample` file is stored is shown below:

```
<INSTALL_DIR>/rcpclient/com.yantra.yfc.rcp_<version>.zip
```

You must extract the contents of this zip file.

To configure locations, you can either create a new `locations.ycfg` XML file or modify the existing `locations.ycfg.sample` XML file.

15.2.2.1 Creating and Configuring a New `locations.ycfg` XML File

To configure a new `locations.ycfg` file:

1. Create the `locations.ycfg` XML file and store it in the `<RCP_EXTN_FOLDER>/resource` directory.

where `<RCP_EXTN_FOLDER>` refers to the folder that you created for storing Sterling RCP-based PCA client application extensions. For more information about creating `<RCP_EXTN_FOLDER>` folder, see [Section 15.2.1, "Creating the RCP_EXTN_FOLDER Folder"](#).

2. Define new locations in the `locations.ycfg` file by using the information provided in the `locations.ycfg.sample` file, which contains proxy server and application server URL settings for various geographical locations.

A sample configuration data from the `locations.ycfg.sample` file is given below:

```
<?xml version="1.0" encoding="UTF-8"?>
<Locations>
  <Location id = "DEFAULT"
    proxyServer="yourproxyserver.com"
    proxyPort="8080"
    updateType = "pull">
    <Config Name = "DEFAULT"
      Protocol = "http"
      BaseUrl = "localhost"
      PortNumber = "7001"
      ApiUrl = "/yantra/RcpServlet"
      CompressionEnabled = "N"
    </Config>
  </Location>
```

```

<Location id = "REMOTE"
    proxyServer="yourproxyserver.com"
    proxyPort="8080"
    updateType ="client">
    <Config Name = "IMAGE"
        Protocol = "http"
        BaseUrl = "localhost"
        PortNumber = "7001"
        ApiUrl = "/icons/rcp/$param1$.gif"
        CompressionEnabled = "N"
    </Config>
</Location>
</Locations>

```

3. Define the Locations root element.
4. Define the Location element under the Locations root element with id such as DEFAULT, LOCAL, REMOTE, and so forth. You can configure the proxy server and application server URL settings for each location.

Note: You must have one Location element with the id attribute value set as "DEFAULT". This Location element must have a Config element whose Name attribute must have the value set as "DEFAULT".

When you log in to a Sterling Rich Client application using a particular location, the system checks whether or not the loaded location has a "DEFAULT" Config element defined for it. If the selected location has the "DEFAULT" Config element, the system loads the "DEFAULT" configuration. Otherwise, the system loads the "DEFAULT" configuration defined in the "DEFAULT" location.

For more information about location configuration settings, see [Section 15.3, "Location Configuration Settings"](#).

15.2.2.2 Modifying the locations.ycfg.sample XML File

To modify the locations.ycfg.sample file:

1. Extract and copy the `locations.ycfg.sample` from `<INSTALL_DIR>/rcpclient/com.yantra.yfc.rcp_<version>.zip` file and store it in the `<RCP_EXTN_FOLDER>/resource` directory.
where `<RCP_EXTN_FOLDER>` refers to the folder that you created for storing Sterling RCP-based PCA client application extensions. For more information about creating `<RCP_EXTN_FOLDER>` folder, see [Section 15.2.1, "Creating the RCP_EXTN_FOLDER Folder"](#).
2. Rename the `locations.ycfg.sample` file to `locations.ycfg` file.
3. Modify the location configurations settings as needed. For information about location configuration settings, see [Section 15.3, "Location Configuration Settings"](#).

15.2.3 Localizing Bundle and Theme Files

You can localize the Sterling Rich Client application's locale-specific files based on the user's locale. The Sterling RCP supports the bundle and theme locale-specific files. All the Sterling Rich Client application plug-ins contain the `<Plug-in_id>_<name>.properties` bundle file and `<Plug-in_id>_<theme_name>.ythm` theme file. For more information about localizing bundle and theme files, see the *Sterling Multi-Channel Fulfillment Solution Localization Guide*.

15.2.4 Enabling HTTPS

If you are using the HTTPS connection to communicate with the application server, copy all SSL (Secure Socket Layer) certificates in the `truststore` directory. For more information about the `truststore` directory, see [Section 15.2.1, "Creating the RCP_EXTN_FOLDER Folder"](#).

For more information about configuring connection settings for HTTPS connection, see [Section 15.4.2, "Configuring Connection Settings for HTTPS Connection"](#).

For a HTTP connection, you can also make selective SSL calls for some sensitive APIs. For more information about configuring connection settings for making selective SSL calls, see [Section 15.4.3, "Configuring Connection Settings for Making Selective SSL Calls"](#).

15.2.5 Applying Updates

The Sterling RCP's update process is based on the timestamp of the files. In the `<INSTALL_DIR>/properties/yfs.properties.in` file, the `yfs.rcp.pca.updates.dir` property points to the directory where updates for the PCAs are located. The `yfs.rcp.pca.updates.cache.dir` property points to the local directory on the application server where updates for the PCAs can be cached.

To deploy updates for the Sterling Rich Client application on a client:

1. Modify the

`<INSTALL_DIR>/properties/customer_overrides.properties` file to configure the following properties:

- Configure the `yfs.yfs.rcp.pca.updates.dir` property by specifying the path of the directory where updates for the PCAs are located. The directory that you specify can also be a shared directory on the network. For example,
`yfs.yfs.rcp.pca.updates.dir =`
`<INSTALL_DIR>/<PCA_UPDATES_DIR>`

where `yfs.rcp.pca.updates.dir` is the property, `<INSTALL_DIR>` is the directory where you have installed Sterling Multi-Channel Fulfillment Solution, and `<PCA_UPDATES_DIR>` is the directory which contains individual updates folder for each Sterling Multi-Channel Fulfillment Solution PCA.

For example, if the root folder for PCA updates is maintained in the `<INSTALL_DIR>/<PCA_UPDATES_DIR>` directory, and for Sterling COM PCA, if the application identifier is `YFSSYS00011`, PCA code is `com20`, and operating system configuration is `win32.win32.x86`, the client searches for updates based on the application ID, PCA code, and operating system configuration. The Sterling COM PCA updates are maintained in the `<INSTALL_DIR>/<PCA_UPDATES_DIR>/YFSSYS00011/com20/win32.win32.x86` directory.

where `C:\Supply_Chain_Apps\pcaupdates` is the value specified in the `yfs.yfs.rcp.pca.updates.dir` property.

You can find the following resources in this directory:

- Sterling RCP client plug-in
- Sterling COM PCA and related plug-ins

- Eclipse related plug-ins

Note: The JRE files are not updated.

- Configure the `yfs.yfs.rcp.pca.updates.cache.dir` property by specifying the path of the local directory on the application server where updates for PCAs need to be cached. For example,
`yfs.yfs.rcp.pca.updates.cache.dir =`
`<INSTALL_DIR>/<PCA_UPDATES_DIR>/<UPDATES_CACHE_DIR>`

Note: Make sure that the directory specified in the `yfs.rcp.pca.updates.dir` property is different from the directory specified in the `yfs.rcp.pca.updates.cache.dir` property.

2. Modify the `locations.ycfg` file to define the type of update you want to deploy on the client in the `updateType` attribute of the `Location` element. The Sterling RCP supports two methods of deploying updates on the client: Client Pull and Push Updates. For more information about the different types of updates that the Sterling RCP supports, see [Section 15.2.5.1, "Type of Updates"](#).

15.2.5.1 Type of Updates

The Sterling RCP's update process is based on the timestamp of files. The Sterling RCP supports two methods of deploying updates for a Rich Client application on the client:

- **Client Pull or Automatic Update**—Client Pull is the automatic way of deploying updates on the client. In this type of update, when a user logs in to a Sterling Rich Client application, based on the location configuration settings, the client application automatically starts searching for updates in a background thread and installs them. Once all updates are downloaded successfully and installed, the user is authorized to restart the application.
- **Push Updates or Manual Update**—Push Updates is the manual way of deploying updates on the client. If you want to use push updates option, copy the contents of the update directory based on the client application which you want to update to the client machine. For

example, if you have specified the update directory as:
`<INSTALL_DIR>/<PCA_UPDATES_DIR>`, copy the contents from the following directory to the client machine.

```
<INSTALL_DIR>/<PCA_UPDATES_DIR>/<PCA_APPLICATION_ID>/<PCA_APPLICATION_VERSION>/<OS_CONFIG>
```

where `<INSTALL_DIR>` refers to the directory where you have installed the Sterling Multi-Channel Fulfillment Solution.

`<PCA_UPDATES_DIR>` is the directory where updates are located,
`<PCA_APPLICATION_ID>` is the identifier of the client application for which you want to deploy updates using the Push Update method,
`<PCA_APPLICATION_VERSION>` is the version number of the client application, and `<OS_CONFIG>` refers to the
`<Windowing_System>.<OS>.<OS_ARCH>` operating system.

15.2.6 Running the Ant Script

Run the application-specific ant script with the appropriate ant target as needed. The ant script is provided by the appropriate Sterling Rich Client application. For example, if you want to deploy the Sterling COM PCA, run the `buildcomapplication.xml` file.

The ant file contains multiple ant targets to generate the deployable folder for all unique combinations of the Operating System and Application such as `buildCOMForWindows`, `buildCOMForLinuxGTK`, `buildSOPForWindows`, and so forth.

For example, if you want to deploy the Sterling COM PCA on Windows, run the following ant script from the `<INSTALL_DIR>` directory with an ant target:

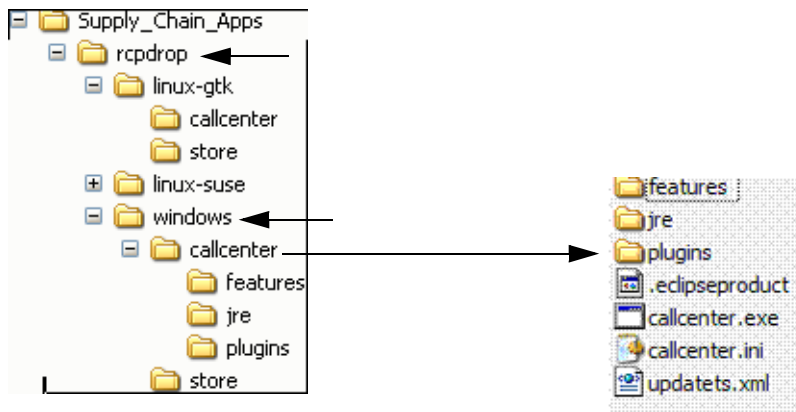
```
ant -f bin/buildcomapplication.xml buildCOMForWindows
```

Note: For this ant script to run, the following arguments or variables need to be exported:

- `<INSTALL_DIR>`—name of the folder where the Sterling Multi-Channel Fulfillment Solution is installed.
- `<RCP_EXTN_FOLDER>`—specify the name of the `<RCP_EXTN_FOLDER>` folder that you created for storing Sterling RCP-based PCA client application extensions. For more information about creating `<RCP_EXTN_FOLDER>` folder, see [Section 15.2.1, "Creating the RCP_EXTN_FOLDER Folder"](#) on page 207.

After you run this ant script runtime the following resources or directory structure is created or generated for the call center application:

Figure 15–2 Sample Directory Structure for Call Center Application



- The `rcpdrop` folder is created within the `<INSTALL_DIR>` directory. where `<INSTALL_DIR>` refers to the directory where you have installed the Sterling Multi-Channel Fulfillment Solution.
- Based on the ant target that you specified, when you run the ant script, a folder for the operating system is created. For example, the `windows` folder is created if you specify `buildCOMForWindows` as the

ant target to deploy the Sterling COM PCA on the Windows operating system.

- Under the windows folder, the <application> folder is created. For example, com.

The com folder contains the required files and resources for the application that are to be built. These resources are accumulated from the following folders:

- <INSTALL_DIR>/rcpclient/
- <INSTALL_DIR>/rcpdependencies/windows
- <INSTALL_DIR>/rcp/COM/rcpclient
- <RCP_EXTN_FOLDER>

where <INSTALL_DIR> refers to the directory where you have installed the Sterling Multi-Channel Fulfillment Solution.

<RCP_EXTN_FOLDER> refers to the folder that you created for the storing Sterling RCP-based PCA client application extensions. For more information about creating the <RCP_EXTN_FOLDER> folder, see [Section 15.2.1, "Creating the RCP_EXTN_FOLDER Folder"](#).

- Also the updatets.xml file is created which is used by the Sterling RCP to check for auto updates.

The updatets.xml file contains a list of files that are present in the application. It also includes the timestamp for these files.

Note: The updatets.xml file is automatically generated by the ant script provided with the Sterling Multi-Channel Fulfillment Solution for building a PCA Application.

15.3 Location Configuration Settings

Location configurations are defined in the locations.ycfg file. You can set different preferences for each location.

To define a new location configuration:

1. Set the attributes of the Location element. For Location element attributes and their descriptions, see [Section 15–1, "Location Element Attribute List"](#).

Table 15–1 Location Element Attribute List

Attribute	Description
id	Specify a unique identifier for the geographical location. For example, DEFAULT, REMOTE, LOCAL, and so forth.
proxyServer	Specify the unique proxy server used to connect to the internet, if applicable.
proxyPort	Specify the port number of the proxy server.
updateType	Set this attribute only when you are updating a Sterling Rich Client application. Specify the mode of update you want to perform, if applicable. Valid update modes are: pull and push. For more information about update modes, see Section 15.2.5, "Applying Updates" on page 215.

2. Define a Config element under the Location element to configure the connection settings. Each location has multiple Config elements. For example, DEFAULT, IMAGE_SMALL, IMAGE_BIG, and so forth. Using the Config element, define the various configuration settings. Set all attributes of the Config element to specify the application server URL you want to use. For more information about configuring connection settings, see [Section 15.4, "Configuring Connection Settings"](#).

Note: You must have one Location element with id attribute value as "DEFAULT" and this Location element must have a Config element whose Name attribute should have the value as "DEFAULT".

When you log into a Sterling Rich Client application using a particular location, the system checks whether or not loaded location has a "DEFAULT" Config element defined for it. If the selected location has "DEFAULT" Config element, the system loads the "DEFAULT" configuration. Otherwise the system loads the "DEFAULT" configuration defined in the "DEFAULT" location.

A sample configuration data used to define a location configuration is as follows:

```
<Location id = "DEFAULT"
```

```

proxyServer="proxy.sterling.com"
proxyPort="8080">
<Config Name = "DEFAULT"
    Protocol = "http"
    BaseUrl = "localhost"
    PortNumber = "7001"
    ApiUrl = "/yantra/RcpServlet"
    CompressionEnabled = "N"
</Config>
</Location>

```

When you start the Sterling Rich Client application, the system reads the `locations.ycfg` file and loads the location information available in this file.

When you start the application for the first time, the Location Preferences window displays.

1. Select a location from the drop-down list.
2. Configure the proxy server settings, if applicable.

Based on the location preferences, you are logged into the application.

15.4 Configuring Connection Settings

To connect to the application server, you must configure the Sterling Rich Client application. In the `locations.ycfg` file, set the protocol, base URL, port number, API URL, and other attributes of the Config element. For Config element attributes, see [Table 15–2](#). You can configure the connection settings for fetching images from the server or connecting to HTTPS.

Table 15–2 Config Element Attribute List

Attribute	Description
Name	Specify a unique name for the server configuration. For example, LOCAL, DEFAULT, and so forth.
Protocol	Specify the name of the protocol to use to communicate with the application server. For example, http or https. For more information about configuring connection settings for HTTPS protocol, see Section 15.4.2, "Configuring Connection Settings for HTTPS Connection" .
BaseUrl	Specify the base URL path of the application server. For example localhost or 10.11.25.80 or www.myserver.com.
PortNumber	Specify the port number based on the protocol you specified. For example, 7001 or 7002.
HttpsPortNumber	(Optional) Specify the port number you want to use to make selective SSL calls for an HTTP connection. For example, 7008. For more information about configuring connection settings for making selective SSL calls, see Section 15.4.3, "Configuring Connection Settings for Making Selective SSL Calls" .
ApiUrl	Specify the URL path of the application server where all APIs are stored. For example, /yantra/RCPServlet. If you want to display images from the server, the URL path must contain \$param1\$ parameter. For more information about configuring connection settings for fetching images from the server, see Section 15.4.1, "Configuring Connection Settings for Fetching Images from the Server" .
CompressionEnabled	If the data received from the server is in the compressed format, set the CompressionEnabled attribute to "Y". The Sterling RCP supports only Gzip compression format. For more information about the supported compression format, see Section 15.7, "Compression in the Sterling Rich Client Platform" .

A sample configuration data used to configure a server is as follows:

```
<Config Name = "DEFAULT"
    Protocol = "http"
    BaseUrl = "localhost"
    PortNumber = "7001"
```



```
        ApiUrl = "/yantra/RcpServlet"
        CompressionEnabled = "Y"
    </Config>
    <Config Name = "LOCAL"
        Protocol = "http"
        BaseUrl = "localhost"
        PortNumber = "7001"
        HttpsPortNumber = "7002"
        ApiUrl = "/yantra/RcpServlet"
        CompressionEnabled = "N"
    </Config>
```

Note: You must have one location element with `id` attribute value as "DEFAULT". This location element must have a `Config` element with `Name` attribute value as "DEFAULT", which defines the DEFAULT URL for the connecting to the application server.

The Sterling Rich Client application is initially launched by connecting to the server specified in the "DEFAULT" URL. You can define the URL at each command level, if applicable. If the command element in the `<Plug-in_id>_commands.ycml` file is not associated with the URL, the system considers the "DEFAULT" URL for that command.

15.4.1 Configuring Connection Settings for Fetching Images from the Server

You can configure the connection settings to fetch images from the server by setting the protocol, base URL, port number, API URL, and other attributes of the `Config` element in the `locations.ycfg` file. For `Config` element attributes, see [Table 15–3](#). You can create more than one configurations to display different types of images.

Table 15–3 Config Element Attribute List

Attribute	Description
Name	Specify a unique name for the server configuration.
Protocol	Specify the name of the protocol to use to communicate with the application server. For example, http or https.

Table 15–3 Config Element Attribute List

Attribute	Description
BaseUrl	Specify the base URL path of the server. For example localhost, 10.11.25.80, or www.myserver.com.
PortNumber	Specify the port number based on the protocol that you have specified. For example, 80.
ApiUrl	Specify the URL path of the server where all the images are stored. The URL path must contain \$param1\$ parameter. For example, /icons/rcp/\$param1\$.gif.
DefaultApiUrl	Specify the URL path of the image that displays if the image specified in the ApiUrl is not found, if applicable. For example. /icons/rcp/404.jpeg.

Note: You can create the following server configurations to fetch images of different types such as GIF, JPEG, PNG, and so forth:

- IMAGE
- IMAGE_SMALL
- IMAGE_MEDIUM
- IMAGE_BIG

Each location must have a server configuration named "IMAGE" which defines the URL for fetching images from the server. You can configure the "IMAGE" URL to get images of type GIF, JPEG, PNG, and so forth. All other server configurations are optional.

The sample configuration data that is used to configure server for displaying images is given below:

```
<Config-List>
  <Config Name = "IMAGE"
    Protocol = "http"
    BaseUrl = "localhost"
    PortNumber = "80"
    ApiUrl = "/icons/imgservlet/?file=$param1$"
  </Config>
```

```

<Config Name = "IMAGE_SMALL"
    Protocol = "http"
    BaseUrl = "localhost"
    PortNumber = "80"
    ApiUrl = "/icons/rcp/$param1$.gif"
    DefaultApiUrl = "/icons/rcp/404.gif"
</Config>
<Config Name = "IMAGE_BIG"
    Protocol = "http"
    BaseUrl = "localhost"
    PortNumber = "80"
    ApiUrl = "/icons/rcp/$param1$.jpeg"
    DefaultApiUrl = "/icons/rcp/404.gif"
</Config>
</Config-List>

```

For example, to get an image from the server using the `http://localhost:80/icons/imgservlet/?file=Y001` URL, define a Config element named `IMAGE` as shown in the sample code (above). To fetch an image from the server using the `http://localhost:80/icons/rcp/Y001.gif` URL, define a Config element named `IMAGE_SMALL` as shown in the sample code (above). In both the cases, the `$param1$` variable is replaced by the image's name.

Note: You can modify the Config element for the IMAGE URL. But ensure that you do not delete it.

For example, if you want to get an image for an OrderNo label:

1. Set the source binding for the label as:

```
lblOrderNo.setSourceBinding("ServerImageList:Images/Icons/RCP/Image1/@OrderNo");
```

where `lblOrderNo` is the label name and `ServerImageList` is the namespace for the model.

2. Set the server image configuration for the label to display the image from the server as shown:

```
lblBindingData.setServerImageConfiguration(YRCConstants.IMAGE_SMALL);
```

where `lblBindingData` is the binding object and `IMAGE_SMALL` is the value of the `Name` attribute of the `Config` element, which is defined in the configuration file.

When getting the image for the `lblOrderNo` label, the `$param1$` parameter is replaced by the value of the `OrderNo` attribute. If the value of the `OrderNo` attribute is "Y001", the image `Y001.gif` displays for the `lblOrderNo` label.

15.4.2 Configuring Connection Settings for HTTPS Connection

To configure the connection settings to communicate with application servers:

1. In the `locations.ycfg` file when defining the connection settings, set the value of `Protocol` attribute of the `Config` element as "https". Also, specify the port number for the HTTPS protocol in the `PortNumber` attribute of the `Config` element. For more information about configuring the connection settings, see [Section 15.4, "Configuring Connection Settings"](#).
2. By default, during handshaking, if there is a mismatch between the URL's hostname and the server's identification hostname, the Sterling RCP allows the HTTPS connection.
3. Copy all SSL or public key certificates required for configuring an HTTPS connection in the `truststore` directory under the `extensions` folder that you created as shown:

```
<RCP_EXTN_FOLDER>/truststore.
```

where `<RCP_EXTN_FOLDER>` refers to the folder that you created for storing the Sterling RCP-based PCA client application extensions. For more information about creating `<RCP_EXTN_FOLDER>` folder, see [Section 15.2.1, "Creating the RCP_EXTN_FOLDER Folder"](#).

A trusted Certificate Authority (CA) like VeriSign issues these security certificates. For more information about SSL or security certificates, see [Section 15.5, "Security Certificates"](#).

15.4.3 Configuring Connection Settings for Making Selective SSL Calls

For a HTTP connection, you can make selective SSL calls for certain sensitive APIs. For example, you use an HTTP connection for calling APIs. For some sensitive APIs you can make an SSL call instead of a normal HTTP call by configuring the connection settings as follows:

- When defining the connection settings in the `locations.ycfg` file, in the `HttpsPortNumber` attribute of the `Config` element, specify the port number you want to use to make selective SSL calls. For more information about configuring the connection settings, see [Section 15.4, "Configuring Connection Settings"](#).
- By default, during handshaking, if there is a mismatch between the URL's hostname and the server's identification hostname, the Sterling RCP allows the HTTPS connection.

Note: You must provide your own custom verification logic by adding the hostname verifier. For more information about adding the hostname verifier, see Section 6.21.1. "Adding the Hostname Verifier" in the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.

- Copy all SSL or public key certificates that are necessary for configuring a HTTPS connection in the `<RCP_EXTN_FOLDER>/truststore` directory that you created.

where `<RCP_EXTN_FOLDER>` refers to the folder that you created for storing the Sterling RCP-based PCA client application extensions. For more information about creating `<RCP_EXTN_FOLDER>` folder, see [Section 15.2.1, "Creating the RCP_EXTN_FOLDER Folder"](#).

A trusted Certificate Authority (CA) such as VeriSign, issues these security certificates. For more information about SSL or security certificates, see [Section 15.5, "Security Certificates"](#).

- Create the `secureapis.xml` file and add the names of all APIs that you want to call by making SSL calls to this file. For more information about adding secure APIs, see [Section 15.6, "Adding Secure APIs for Making Selective SSL Calls"](#).

- Copy the `secureapis.xml` file to the `<RCP_EXTN_FOLDER>/resources` directory that you created.

15.4.4 Configuring Connection Settings for Online Help

To access the online help, configure the connection settings by setting the protocol, base URL, port number, and API URL attributes of the `Config` element in the `locations.ycfg` file. For `Config` element attributes, see [Table 15–3](#).

The following sample configuration data can be used to configure the connection settings for accessing the online help:

```
<Config Name = "HELP"
    Protocol = "http"
    BaseUrl = "localhost"
    PortNumber = "80"
    ApiUrl =
        "/yantradocs/yfscommon/online_help/$param1/wwhelp/wwhimpl/c
        ommon/html/wwhelp.htm?context=$param2_userguide&topic=$
        param3"
</Config>
```

Here `$param1` refers to locale, `$param2` refers to module id, and `$param3` refers to help topic id.

15.5 Security Certificates

An SSL certificate or public key certificate is a certificate that uses a digital signature to bind a public key with an identity information such as the name of the person or an organization, address, and so forth. An SSL certificate has information about the owner of the certificate, the usage of the certificate, validity details, resource location or web site address, e-mail address and the certificate ID of the person who certifies (signs) this information. SSL certificates are used for secure communication over the HTTPS protocol.

Whenever a client needs to verify the authenticity of an SSL server, the SSL certificate used by the server needs to be signed by the Certificate Authority that is already trusted by the client. The well-known certificate authorities such as Thawte and VeriSign serve as an authoritative, trusted third party for authentication. They sign the SSL certificates that are used when dealing with sensitive information or services. If these

SSL certificates are signed by a trusted authority, it is possible to verify the identity of a server by supplying the SSL certificate.

15.6 Adding Secure APIs for Making Selective SSL Calls

First create the `secureapis.xml` file. To add secure APIs to this file, follow these steps:

1. In the `secureapis.xml` file, define the `Apis` root element.
 - In an application, if you always want to call a specific API by making an SSL call, define the `Api` element under the `Apis` root element.

In the `name` attribute of the `Api` element, specify the name of the API for which you want to make the selective SSL call.

For example, in an application you may always want to call the `getCreditCardDetails` API using an SSL call.

- In an application, for some specific screens, if you want to call a particular command by making an SSL call, define the `Command` element under the `Apis` root element.

In the `commandName` attribute, specify the name of the command for which you want to make an SSL call. In the `formId` attribute, specify the identifier of the form for which you are calling the command specified in the `commandName` attribute.

For example, in a particular screen, you may want to call the `getUserVerification` command by making a secure SSL call.

A sample `secureapis.xml` file is shown here:

```
<?xml version="1.0" encoding="UTF-8"?>
<Apis>
  <Api name="getCreditCardDetails"/>
  <Command formId="com.yantra.pca.ycd.rcp.YCDUserDetails"
    commandName="getUserVerification"/>
</Apis>
```

Note: You can create the `secureapis.xml` file and add the secure APIs to this file using the Sterling RCP Extensibility Tool. For more information about using the Sterling RCP Extensibility Tool, see the *Sterling Multi-Channel Fulfillment Solution Customization Guide*.

15.7 Compression in the Sterling Rich Client Platform

The Sterling RCP enables you to send and receive compressed data to and from the application server. When you enable compression, the Sterling RCP enables bidirectional compression.

Benefits

- The bidirectional compression helps in reducing the traffic in both directions as only the XML data is passed to an API or service. For example, input XMLs and output templates passed to an API or service.
- The compression is most useful for applications that rely more on multiple API calls because it avoids multiple trips to and from the application server.
- There is minimal overhead in performing compression. For example, when an XML file size is large, we can reduce the size of the data by about 90%.

Note: The Sterling RCP supports the Gzip compression format.

The Sterling RCP does not support compression of images or zip files when fetching images or extracting updates from the server.

To enable compression, in the `locations.ycfg` file, you must set the value of the `CompressionEnabled` attribute of the `Config` element to "Y". These settings are done when you are configuring the connection settings for a Sterling Rich Client application. For more information about

configuring connection settings, see [Section 15.4, "Configuring Connection Settings"](#).

A

Property Files Reference

The following property files are described in this chapter:

- [yfs.properties*](#) - see page 234
- [dbclassCache.properties.*](#) - see page 258
- [Sandbox.cfg](#) - see page 259

Warning: Sterling Commerce supports changes to the properties included in this chapter only. Changes to any other properties are not supported.

A.1 Making Changes to Properties

Do not directly edit or change the property files covered in this appendix, except `sandbox.cfg`. To make changes to the properties in these files, you must use the `customer_overrides.properties` file or `sandbox.cfg`. Sterling Commerce does not recommend that you modify or change any properties in files ending with `.in` because newer versions or patches of the product will overwrite your changes. Sterling Commerce also does not recommend that you change a property file that has a corresponding `.in` file because the `setupfiles` script will re-create the properties file again, thus causing you to lose your changes.

For detailed information about using the `customer_overrides.properties` file and `sandbox.cfg`, see [Chapter 12, "Configuring Properties"](#).

A.2 Setting the Database Connection Properties

Database connection properties are configured during installation. They are placed in the <INSTALL_DIR>/properties/sandbox.cfg file and are used to set values in the jdbc.properties file when the setupfiles utility is run.

The DB_SCHEMA_OWNER is set by the installer to the default schema of the user for whom the installation is being run. If you want to put objects in a schema other than the default schema for Oracle or for DB2, you must edit the sandbox.cfg file and change the DB_SCHEMA_OWNER variable to the desired value. The system will use the variable to decide what schema to work against.

A.3 yfs.properties*

The yfs.properties.* files contain business-level properties. The following table describes the properties that can be changed. The properties are grouped in the following categories:

- [Agent](#) - see page 235
- [Analytics](#) - see page 238
- [Database](#) - see page 239
- [Exception Management](#) - see page 240
- [Implementation](#) - see page 240
- [Inventory Management](#) - see page 242
- [JMS](#) - see page 243
- [Online Help](#) - see page 243
- [Order Management](#) - see page 244
- [Parcel Carrier Server](#) - see page 244
- [Prints](#) - see page 246
- [Security](#) - see page 246
- [Service Definition Framework \(SDF\)](#) - see page 248
- [System Management](#) - see page 250
- [User Interface](#) - see page 251

- [Warehouse Management](#) - see page 252

Property	Values	Description
Agent		
yfs.agent.override.providerurl	Default is not set.	<p>AgentServer Override for provider url: set the property to the provider URL which will be used by ALL Agent servers. This property does not impact integration servers configured in the service builder. This property overrides the url configured for time triggered transactions in config.</p> <p>Example for Weblogic:</p> <pre>yfs.agent.override.providerurl=t3://<host>:<port></pre> <p>Example for WebSphere:</p> <pre>yfs.agent.override.providerurl=corbaloc::<host>:<bootstrapport></pre> <p>Example for JBoss:</p> <pre>yfs.agent.override.providerurl=jnp://<ipaddress>:<port></pre>
yfs.agent.override.icf	Default is not set.	<p>AgentServer Override for InitialContextFactory Name: set the property to the InitialContextFactory Name which will be used by ALL Agent servers. This property does not impact integration servers configured in the service builder. This property overrides the InitialContextFactory configured for time triggered transactions in config.</p> <p>Example for WebSphere:</p> <pre>yfs.agent.override.icf=com.ibm.websphere.naming.WsnInitialContextFactory</pre> <p>Example for WebLogic:</p> <pre>yfs.agent.override.icf=weblogic.jndi.WLInitialContextFactory</pre> <p>Example for JBoss:</p> <pre>yfs.agent.override.icf=org.jnp.interfaces.NamingContextFactory</pre>

Property	Values	Description
yfs.agent.override.qcf	Default is not set.	<p>AgentServer Override for QueueConnectionFactory Name: set the property to the QueueConnectionFactory Name which will be used by ALL Agent servers. This property does not impact integration servers configured in the service builder. This property overrides the QueueConnectionFactory configured for time triggered transactions in config.</p> <p>Example:</p> <pre>yfs.agent.override.qcf=<QueueConnectionFactory Name></pre>
yfs.agent.override.retryCount	Default is not set.	<p>AgentServer Override for RetryCount: set the property to the JMS RetryCount to be used by ALL Agent Servers. This property does not impact integration servers configured in the service builder.</p> <p>Example:</p> <pre>yfs.agent.override.retryCount=<Number of Retries></pre>
yfs.agent.override.retryInterval	Default is not set.	<p>AgentServer Override for RetryInterval: set the property to the JMS RetryInterval to be used by ALL Agent Servers. This property does not impact integration servers configured in the service builder. This interval is specified in milliseconds.</p> <p>Example:</p> <pre>yfs.agent.override.retryInterval=<RetryInterval in Milliseconds></pre>

Property	Values	Description
yfs.agent.backup.providerurl	Default is not set.	<p>AgentServer Backup for JMS. Set these three properties to the JMSprovider URL InitialContextFactory Name and QueueConnectionFactory Name which will be used by ALL Agent servers as a Backup if the primary JMSServer becomes unavailable, the QueueName used by the Agent Server on the backup JMSServer will be the same as the one configured on the primary JMSServer.</p> <p>These properties will be used only after a successful start of the Agent Server using the primary JMSServer configuration. If the AgentServer Override properties are specified, they are used as the primary JMSServer properties. The backup JMSServer will be used by the Agent Server only if all three properties are specified.</p> <p>These properties do not impact integration servers configured in the service builder.</p> <p>Examples for WebSphere:</p> <pre>yfs.agent.backup.providerurl=corbaloc::<host>:<bootstrapport> yfs.agent.backup.icf=com.ibm.websphere.naming.WsnInitialContextFactory</pre> <p>Examples for WebLogic:</p> <pre>yfs.agent.backup.providerurl=t3://<host>:<port> yfs.agent.backup.icf=weblogic.jndi.WLInitialContextFactory</pre> <p>Examples for JBoss:</p> <pre>yfs.agent.backup.providerurl=jnp://<ipaddresses>:<port> yfs.agent.backup.icf=org.jnp.interfaces.NamingContextFactory</pre>
yfs.agent.backup.icf	Default is not set.	
yfs.agent.backup.qcf	Default is not set.	
yfs.agent.backup.retryCount	Default = 0	<p>Number of times to retry a failed JMS connection</p> <p>Example: yfs.agent.backup.retryCount=3</p>
yfs.agent.backup.retryInterval	Default = 0	<p>Number of milliseconds to wait between retries</p> <p>Example: yfs.agent.backup.retryInterval=3</p>
yfs.agent.override.auth.enabled	Valid values = Y/N	<p>The agent overrides to override the JMS Security parameter values specified in the Agent criteria. If set to Y, you must also include values for the following two properties (userid and password).</p> <p>Example: yfs.agent.override.auth.enabled=Y</p>

Property	Values	Description
yfs.agent.override.auth.userid	Default is not set.	If yfs.agent.override.auth.enabled is set to Y, these properties (userid and password) must be present, otherwise an error is thrown.
yfs.agent.override.auth.password		
yfs.agentserver.queryTimeout		<p>Set this property to specify the Query Time out for Agents. This is the global property applicable for all agents.</p> <p>It is possible for individual agents to override this property by specifying their own query timeouts. For example, the User Activity Audit Purge Agent can specify a property like, USERACTAUDITPRG.queryTimeout. This will override this global property. Set to zero means unlimited.</p> <p>Example: <code>yfs.agentserver.queryTimeout=</code></p>
Analytics		
analytics.portal.url		<p>Powerplay Launch URL. Uncomment the line below and replace the <machine> with the Cognos Upfront installation machine name/ip address. Do not change anything else in the URL, else Analytics access will fail.</p> <p>Example:</p> <pre>http://<machine>/cognos/cgi-bin/login.cgi?session=#USERID#&password=#PASSWORD#&return_url=upfcgi.exe</pre>
analytics.reportnet.url		<p>Reportnet Analytics Launch URL. Uncomment the line below and replace the <machine> with the Cognos ReportNet installation machine name/ip address. Do not change anything else in the URL, else ReportNet access will fail.</p> <p>Example:</p> <pre>http://<machine>/crn/cgi-bin/cognos.cgi</pre>
analytics.namespace		<p>Analytics namespace. Indicates the namespace that has been configured in COGNOS ReportNet authentication, against which the users would be authenticated.</p> <p>Example: <yantra></p>
yfs.analytics.checkCachedReports		<p>Set this to Y to view the cached reports from Yantra Analytics Console.</p> <p>Example: <code>yfs.analytics.checkCachedReports=N</code></p>

Property	Values	Description
Database		
yfs.dblogin.yantraschema.name	<SchemaName> Default is not set.	The database schema for the Application installation (if different from the userid being used) Example: yfs.dblogin.yantraschema.name=<SchemaName>
yfs.cursor.sharing.mode.dcm	Default = FORCE	Note: This property is required only when the database is Oracle and you are integrating with Yantra DCS. Set this to the current value of cursor_sharing parameter in your database. You can find it in v\$parameter table of oracle. Example: yfs.cursor.sharing.mode.dcm=FORCE
yfs.dblogin.datasource.name	The value will depend on how configuration is done by the application server administrator. Default is not set (that is, the default is not using a datasource).	To enable database connection pooling, create the pool, configure the data source entry for JNDI, and specify it here. (Refer to your application server vendor's manuals for details on connection pool management.)
yfs.db.textsearch	Valid values = Y, N Default = N	Setting this property will decide whether Text Search is to be enabled. Example: yfs.db.textsearch=N
yfs.db.textsearch.oracle.contexttype	Valid values = CTXCAT, CONTEXT Default = CTXCAT	Set this property to determine the type of text index(ctxcat/context) in oracle. Example: yfs.db.textsearch.oracle.contexttype=ctxcat
SUFFIX_KEY_WITH_INSTANCE_NO	Valid values = Y, N Default = N	Set this property to insert instance number into primary keys generated. Example: SUFFIX_KEY_WITH_INSTANCE_NO=N
yfs.enable.proxy.sql.logging		These three properties help in obtaining logs for SQL logging. If the application server is running on a Windows environment use the directory delimiter as "/" for the proxy log directory because only UNIX delimiter ("/") will be recognized. Examples: yfs.enable.proxy.sql.logging=Y yfs.enable.source.logging=Y yfs.proxy.log.dir=&INSTALL_DIR;/logs/sqlproxylogs
yfs.enable.source.logging		
yfs.proxy.log.dir		

Property	Values	Description
Exception Management		
yfs.onerror.raisealert	Valid values = Y/N Default = Y	When this property is turned on (set to Y), errors encountered are directed to the alert console. Errors are grouped by certain criteria. Not every error is a separate entry in the alert console. Example: <code>yfs.onerror.raisealert=Y</code>
yfs.onerror.raisealert.logging.interval	Valid values = day/hour Default = day	This property is to set the logging interval. If it is set to day, errors are grouped on daily basis. Otherwise, they are grouped on an hourly basis. If an error is repeated within a logging interval, it is not directed multiple times to alert console. Instead, the alert entry shows an increased occurrence count. Example: <code>yfs.onerror.raisealert.logging.interval=day</code>
yfs.onerror.raisealert.expiration.days	Number of days Default = 7 A value of 0 means the alert will never be closed by this agent.	This property controls how many days of inactivity are required before the alert is eligible to be automatically closed by the Inbox Purge Agent. Example: <code>yfs.onerror.raisealert.expiration.days=7</code>
yfs.onerror.raisealert.queuekey	<QUEUE_KEY> Default = DEFAULT	Set this to the QUEUE_KEY of the Queue the alert should be assigned to. Example: <code>yfs.onerror.raisealert.queuekey=</code>
Implementation		
yfs.comsupport	Valid values = Y, N Default = Y	Indicates whether or not your system supports COM. If COM is enabled, you can configure actions to call COM objects. Example: <code>yfs.comsupport=Y</code>
yfs.purge.path	Default is &APP_DIR;/logs	This property must be set for the purge programs to run. It contains the absolute path to the directory where purge logs will be written. Example: <code>yfs.purge.path=&APP_DIR;/logs</code>
log4j.configuration	<Property Name> Default = /resources/log4jconfig.xml	Property to handle logging. This property points to the location of the log4j configuration xml file. Example: <code>log4j.configuration=/resources/log4jconfig.xml</code>
yfs.install.localecode	<locale code> Set to en_US_EST when the Sterling Multi-Channel Fulfillment Solution is delivered; If changes are necessary, this must be changed by the customer.	Installation locale code. This localecode has to match the underlying OS default timezone where the database is installed to avoid timezone calculation errors. Example: <code>yfs.install.localecode=en_US_EST</code>

Property	Values	Description
yfs.install.displaydoublequantity	Valid values = Y/N Default = Y	This property should be set to "Y" if you want to support fractional quantities for attributes which belong to QUANTITY datatype. Example: yfs.install.displaydoublequantity=Y
yfs.file.encoding	<Property name> Default = UTF-8	This property controls the encoding of the files, like API template, theme XMLs, exception template (except e-mail templates which is controlled by yfs.email.template.encoding). Example: yfs.file.encoding=UTF-8
yantra.app.maxrecords	<number of records> Default = 5000	This property sets the default number of records returned by Application list APIs. Increase the application server JVM heap settings if these parameters are increased. Change will affect search limits for all users. Example: yantra.app.maxrecords=5000
yfs.app.identifyconnection	Valid values = Y/N Default = N	Set this property to "Y" to enable the application to set contextual information (Agent, API name) on the connection. The information stamped on the connection can be viewed on the database connection by the tools provided by the database vendors. This allows mapping of the connection in the database to its origin on the application side. Note: This is available only for Oracle and DB2. Example: yfs.app.identifyconnection=Y
yfs.uidev.refreshResources	Valid values = Y/N Default = N	This property can be set to "Y" when developing and customizing the Application Console UI. When set to "Y", refresh actions will be available within the Resource Configurator screens. These actions can be used to refresh the resources used in the Console UI Framework without having to restart the application server. If this property is not set to "Y", then the actions will be disabled, and it will be necessary to restart the application server whenever any resource is changed within the configurator. IMPORTANT: This property should NOT be set to "Y" for live production systems because the refresh only works for a single user development environment. Example: yfs.uidev.refreshResources=N
yantra.document.isnamespaceaware	Valid values = Y/N Default = N	This property should be set to Y to handle namespaces in XML. Example: yantra.document.isnamespaceaware=N

Property	Values	Description
com.yantra.ycp.em.server.taskpollingtime	<number of seconds> Default = 60	Set this property (in seconds) to indicate the interval at which tasks need to be polled for mobile operators. Example: com.yantra.ycp.em.server.taskpollingtime=60
OverrideTransitTime	Valid values = Y/N Default = N	If both Request Delivery date and Request Ship date are passed, setting this property will compute the transit time as difference between these two dates, and no further transit time computation will be required. Example: OverrideTransitTime=N
yfs.install.applyshipdateoptimization	Valid values = Y/N	Set this property to "N" if allocation should not optimize releases based on ReqShipDate. Example: yfs.install.applyshipdateoptimization=Y
yfs.audit.user.session.activity	Valid values = Y, N Default = Y (audit enabled)	Set this property to enable auditing and disable auditing. Example: yfs.audit.user.session.activity=Y
Inventory Management		
yfs.inventory.sortandlock	Valid values = Y/N	Setting this property to Y will prevent dead locking on YFS_Inventory_Item table during order creation and when synchronizing changes from a shipment back to the order. This parameter is only used if the order contains a logical kit, or when a shipment contains shipment lines from different orders. Setting this property to Y will result into holding lock on YFS_Inventory_Item table for longer period. So, this may degrade overall system performance. Because of this reason, you should set this property to Y only if you are getting excessive dead locks on YFS_Inventory_Item table. Example: yfs.inventory.sortandlock=
yfs.hotsku.useHotSKUFeature	Valid values = Y/N Default = N	Set this property to "Y" to use the Hot SKU feature. For more information about the Hot SKU feature, please see Product Concept Guide and the Performance Management Guide. Example: yfs.hotsku.useHotSKUFeature=N
yfs.hotsku.useTimeOutLocking	Valid values = Y/N Default = N	Set this property to "Y" to use locking timeout while trying to obtain a lock for an inventory item when the Hot SKU feature is enabled. The timeout period will be based on the yfs.hotsku.secondsToClassifyAsAbnormalTime property (round up to integer). Example: yfs.hotsku.useTimeOutLocking=N

Property	Values	Description
JMS		
yantra.jms.receive.timeout	<number of milliseconds> Default = 1800000 milliseconds (30 minutes)	Specifies the JMS receive timeout value (in milliseconds). The JMS Receiver receives the next message that arrives within the specified timeout interval. The receive call blocks until a message arrives, the timeout expires, or this message consumer is closed. A timeout of zero never expires, and the call blocks indefinitely. Example: yantra.jms.receive.timeout=1800000
yfs.jms.session.disable.pooling	Valid values = Y/N Default = N	To disable JMS Session pooling, set this property to Y. Sterling recommends setting to N for performance reasons. Example: yfs.jms.session.disable.pooling=N
yfs.flow.override.auth.enabled yfs.flow.override.auth.userid yfs.flow.override.auth.password	Valid values = Y/N	The flow overrides to override the JMS Security parameter values specified in the service definition framework. If this value is set to Y then the other two properties at left (userid and password) must be present, otherwise an error is thrown. Examples: yfs.flow.override.auth.enabled=Y yfs.flow.override.auth.userid=Y yfs.flow.override.auth.password=Y
yfs.jms.session.disable.pooling	Valid values = Y/N Default = N	To disable JMS Session pooling, set this property to Y. By default this property is set to N. Yantra recommends N value to this property for better performance reasons. Example: yfs.jms.session.disable.pooling=N
Online Help		
yfs.urlforhelp.path	</link to online help/> Default = /wwhelp/wwhimpl/common/html/wwhelp.htm	This controls where the link to the online help will go. The value must begin and end in a forward slash (/) to ensure proper behavior. Example: yfs.urlforhelp.path=/wwhelp/wwhimpl/common/html/wwhelp.htm
yfs.heightforhelp.path	<height in pixels> Default = 590	This controls height for the online help. Example: yfs.heightforhelp.path=390
yfs.widthforhelp.path	<width in pixels> Default = 715	This controls width for the online help. Example: yfs.widthforhelp.path=515

Property	Values	Description
yfs.onlinehelp.path	<code></online help path/></code> Default = <code>/<appname>docs/yfscommon/online_help/</code>	This controls where the link to the online help will go. The value must begin and end in a forward slash (/) to ensure proper behavior. It must be a simple path; a full URL will not work. Example: <code>yfs.onlinehelp.path=<appname>docs/yfscommon/online_help/</code>
Order Management		
yfs.transferPreparation.minTimeReq	<code><number of hours></code> Supported maximum value = 24 HRs Default = 0	Set this property with the minimum time (Hours) required for transfer preparation. This property is used by all Promising and Scheduling APIs. Example: <code>yfs.transferPreparation.minTimeReq=0</code>
Parcel Carrier Server		
ycs.airborne.server.url	<code><server url></code>	Airborne logon parameter. Example: <code>ycs.airborne.server.url=https://eCommerce.airborne.com/ApiLandingTest.asp</code>
ycs.airborne.server.UserID	<code><userid></code>	Airborne logon parameter. Example: <code>ycs.airborne.server.UserID=<your airborne server userid></code>
ycs.airborne.server.Passwd	<code><password></code>	Airborne logon parameter. Example: <code>ycs.airborne.server.Passwd=<your airborne server password></code>
ycs.airborne.LabelPrint.X-coordinate	<code><X-coordinate></code>	Airborne print parameter. Example: <code>ycs.airborne.LabelPrint.X-coordinate=0.1</code>
ycs.airborne.LabelPrint.Y-coordinate	<code><Y-coordinate></code>	Airborne print parameter. Example: <code>ycs.airborne.LabelPrint.Y-coordinate=0.1</code>
ycs.airborne.LabelPrint.Width	<code><LabelPrint.Width></code>	Airborne print parameter. Example: <code>ycs.airborne.LabelPrint.Width=8.5</code>
ycs.airborne.LabelPrint.Height	<code><LabelPrint.Height></code>	Airborne print parameter. Example: <code>ycs.airborne.LabelPrint.Height=6.5</code>
ycs.airborne.LabelPrint.Dump_Switch	Valid values = Yes/No	Airborne print parameter. Example: <code>ycs.airborne.LabelPrint.Dump_Switch=yes</code>

Property	Values	Description
ycs.connectship.server.url	<IP address>	<p>Change the IP address according to your ConnectShip integration IP.</p> <p>Example:</p> <pre>ycs.connectship.server.url=http://127.0.0.1:100/Prologistics/XML_Processor/Server/XMLProcDLL.asp</pre>
ycs.fedex.powershipserver.ipaddress	<IP address>	<p>Change the IP address according to your Powership Server IP.</p> <p>Examples:</p> <pre>ycs.fedex.powershipserver.ipaddress=127.0.0.1</pre>
ycs.fedex.powershipserver.portno	<port>	<pre>ycs.fedex.powershipserver.portno=2000</pre>
yfs.log.logger	<absolute path to message log dir>	<p>This property must be set if you use the default message handling implementation. It contains the absolute path to the directory where message logs will be written.</p> <p>Example: yfs.log.logger=</p>
yfs.log.directory	<full path to log dir>	<p>Example:</p> <pre>ycs.xmlDump.directory=<full path of log directory></pre>
yfs.xmlDump.directory		
ycs.timer.switch	Valid values = yes/no	<p>This property is to set to enable and disable the Timer from logging.</p> <p>Example: ycs.timer.switch=yes</p>
ycs.log.size.maxallowed	<max log file size in bytes> Default = 1000000	<p>This property is only used if you use the default message handling implementation. It specifies the maximum allowed log file size in bytes. A new active log file is created whenever the currently active log file reaches this size.</p> <p>Example: ycs.log.size.maxallowed=1000000</p>
ycs.log.logger	<class name>	<p>Default class to handle messages output by PureEcommerce. The default implementation will write these messages to log files. Refer to the Programming Guide for how to write your own implementation for this class.</p> <p>Example:</p> <pre>ycs.log.logger=com.yantra.ycs.util.YCSFileLogger</pre>
ycs.purge.path	<full path of log directory>	<p>This property must be set for the purge programs to run.</p> <p>Example:</p> <pre>ycs.purge.path=<full path of log directory></pre>

Property	Values	Description
yfs.purge.days	<number of days>	This property must be set for the purge programs to run. Example: yfs.purge.days=15
yfs.weight_tolerance_percent	<percent> Default = 5	If a label is printed before it is actually manifested, weight printed on the label may not exactly match the actual weight. Parameter below specifies the tolerance percent beyond which the label has to be reprinted. Example: yfs.weight_tolerance_percent=5
proxySet=false	Valid values = true/false Default = false If set to true, the following details are required: <ul style="list-style-type: none"> https.proxyHost https.proxyUser https.proxyPassword https.proxyPort 	Proxy settings for https connections. Example: proxySet=false
Prints		
yfs.loftware.tcpip.sockets	Valid values = Y/N Default = N	Set this property to Y to enable printing via TCP/IP Socket Interface. Set it to N to enable printing via the File Interface (file "drop" to a shared network drive). Note: Although the default value is set to N for the sake of backward compatibility, our recommendation is to set this flag to Y. Example: yfs.loftware.tcpip.sockets=N
yfs.loftware.tcpip.sockets.mode	Valid values = WAIT, NOWAIT Default = NOWAIT	Set this property to WAIT to change the Software print server's mode to WAIT mode Example: yfs.loftware.tcpip.sockets.mode=NOWAIT
Security		
yfs.login.singlesignon.class	<class name>	The class that handles Single Signon. Refer to the javadocs for the com.yantra.ycp.japi.util.YCPSSOManager interface for information about how to write your own implementation for this class.
yfs.security.singlesignon.enabled	Valid values = Y/N Default = Y	If this property is set to Y, the single sign on class is called.

Property	Values	Description
yfs.login.singlesignon.checkuser	Valid values = Y/N Default = N	If this property is set to Y, each request is validated against the singleSignOn Server for the user authentication. If the property is set to N, user authentication against the singleSignOn Server is done only when the session times out.
yfs.security.authenticator	<class name> Default is not set.	The class that will be invoked for user authentication. Uncomment and change this only if you <u>do not</u> want to use application authentication. Refer to the javadocs for the YFSAuthenticator interface for information about how to write your own implementation for this class. If you want to use the default implementation for LDAP authentication, uncomment the property and set it to com.yantra.yfs.util.YFSLdapAuthenticator. Example: yfs.security.authenticator=
yfs.security.ldap.factory	<class name>	The LDAP context factory classname as specified in your LDAP Server configuration. Example: yfs.security.ldap.factory=com.sun.jndi.ldap.LdapCtxFactory
yfs.security.ldap.url	<url>	The URL for accessing your LDAP Server as specified in your LDAP Server configuration. Example: yfs.security.ldap.url=ldap://<ldapservename>:<portnum>
yfs.security.ldap.ou		The value specified for the organizational unit in your LDAP Server configuration. Example: yfs.security.ldap.ou=
yfs.security.ldap.o		The value specified for the organization in your LDAP Server configuration. Example: yfs.security.ldap.o=
yfs.htmlencoding.triggers	<value 1,value 2,value 3,value n> Default = <,>,),]	This property contains a comma-separated list of characters that could signify potentially unsafe HTML content. If a information being written to the browser is found to contain any of these characters, the output is safely encoded to prevent exploitation of cross-site scripting vulnerabilities in the application. Example: yfs.htmlencoding.triggers=<,>,),]

Property	Values	Description
yfs.encrypter.class	<class name>	The class that handles encryption and decryption of credit card numbers. Refer to the javadocs for the YCPDecrypter interface for information about how to write your own decrypter class. If this property is not specified, then no encryption will be performed.
yfs.propertyencrypter.class		<p>This class will be used for encrypting and decrypting properties specified in yfs.properties, yiclient.properties and management.properties files. All properties which end with ".encrypted" are automatically decrypted using this class at runtime. use this property to encrypt critical data like user/password.</p> <p>Refer to the javadocs for the YCPDecrypter interface for information about how to write your own property encrypter class.</p> <p>Example:</p> <pre>yfs.agent.override.auth.password=<password></pre> <p>could be specified as:</p> <pre>yfs.agent.override.auth.password.encrypted=<encrypted password></pre>

Service Definition Framework (SDF)

yfs.smtp.session.reaptime	Default = 10*60 seconds	This is the frequency at which the JMS/SMTP connection reaper thread will examine the connection pool for connections that are eligible to be closed. The connection will be eligible to be closed if the connection is idle for the number of seconds specified for the reaptime. This property is specified in seconds.
yfs.jms.session.reaptime		
yfs.smtp.connectionpool.enable	Valid values = true/false Default = true	<p>To disable SMTP connection pooling, set this property to false. Sterling Commerce recommends setting this property to true for performance reasons.</p> <p>Example: <code>yfs.smtp.connectionpool.enable=true</code></p>
yfs.emailer.class		<p>The class that handles automated e-mail communication to and from this application.</p> <p>Example:</p> <pre>yfs.emailer.class=com.yantra.util.YFCEmailerImpl</pre>
yfs.email.template.encoding	<encoding type> Default = UTF-8	<p>Handles encoding for e-mail communication to and from this application.</p> <p>Example: <code>yfs.email.template.encoding=UTF-8</code></p>

Property	Values	Description
yfs.xml.uriresolver	<class name>	<p>This class can be used to provide a custom URIResolver during XSL processing. Refer to the Java documentation for more information on this interface.</p> <p>If this property is not present, then the default implementation provided by the XSL processor is used. If set to <code>com.yantra.interop.util.YantraDefaultURIResolver</code>, an attempt is made to resolve the URI within the classpath. If set to another class name, this class is used in place of the <code>YantraDefaultURIResolver</code>. The class given here must implement the <code>javax.xml.transform.URIResolver</code> interface.</p> <p>Example:</p> <pre>yfs.xml.uriresolver=com.yantra.interop.util.YantraDefaultURIResolver</pre>
yfs.sync.jms.request.timetolive	<p><number of seconds></p> <p>Default = responseTimeout value of the receiver</p>	<p>Request queue timetolive in seconds for Synchronous JMS messages. Use this property to set the timetolive value for Synchronous JMS Messages. Note that, this property is a global level override and is applicable for all the services using Synchronous JMS.</p> <p>WARNING: If this property value is set beyond the responseTimeout value, it may result in unprocessed request messages in the reply/response queue. By default, in order to clean up the un-processed messages from the reply/response queue, the timeToLive for request messages is set to the responseTimeout value of the receiver.</p> <p>Example: <code>yfs.sync.jms.request.timetolive=</code></p>
yfs.sync.jms.request.timetolive.<SERVICE_NAME>	<p><number of seconds></p> <p>Default = responseTimeout value of the receiver</p>	<p>Request queue timetolive in seconds for Synchronous JMS messages. Use this property to set the timetolive value for Synchronous JMS Messages. Note that, this property is applicable only for the specified service.</p> <p>WARNING: If this property value is set beyond the responseTimeout value, it may result in un-processed request messages in the reply/response queue. By default, in order to clean up the un-processed messages from the reply/response queue, the timeToLive for request messages is set to the responseTimeout value of the receiver.</p> <p>Example:</p> <pre>yfs.sync.jms.request.timetolive.<SERVICE_NAME>=</pre>

Property	Values	Description
yfs.reprocess.MaxErrorMessageLength	Valid value = integer Default = 4000	Setting this property will decide the length to which the error message to be stored in YFS_REPROCESS_ERROR table should be truncated. Example: <code>yfs.reprocess.MaxErrorMessageLength=</code>

System Management

Note: The next four properties are for the Health Monitor. The health monitor will raise the configured alerts when the api/appServer response time or the agent Pending Job count crosses the threshold.

yantra.hm.api.threshold	<number of milliseconds> Default = 20000 milliseconds	api threshold in milliseconds, this is the system default value used for average response time for appservers. If this value is not specified 20000 milliseconds will be assumed.
yantra.hm.agent.threshold	<job size> Default = 10000	System default value used for the pending jobs size for agents. If this value is not specified, 10000 will be assumed.
yantra.hm.appserver.threshold	<number of milliseconds> Default = 8000 milliseconds	Appserver threshold in milliseconds, this is the system default value used for average response time for api's/services. If this value is not specified 8000 milliseconds will be assumed.
yantra.hm.purge.interval	<number of days> Default = 30 days	Health Monitor purge interval, this is the system default value used for purging heartbeat and snapshot records. if this value is not specified 30 days purge interval is assumed.
yantra.statistics.collect	Valid values = n, N, y, Y Default = N	Property to turn on Statistics logging. Example: <code>yantra.statistics.collect=Y</code>
yantra.statistics.persist.interval	Valid values for minutes (M/m) = 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30 Valid values for minutes (H/h) = 1, 2, 3, 4, 6, 8, 12	Property to determine statistics logging time interval. Format of the property is Xm or Xh where X is an integer between 1 and 60 and M/m for minutes or H/h for hours. If any unrecognized value or unit is specified, it will default to 10m (minutes). If a value of 61m or greater is specified, it will be reduced to 60m. If a value of 25h or greater is specified, it will be reduced to 24h. If the units are minutes (M/m), then the value is rounded up or down to the nearest equal divisor of 60 minutes. If the units are hours (H/h), then the value is rounded up or down to the nearest equal divisor of 24 hours. Example: <code>yantra.statistics.persist.interval=10m</code>

Property	Values	Description
User Interface		
yfs.disable.webpages.caching	Valid values = Y, y, N, n Default = N	Setting this property to 'Y' will disable caching of Application pages. Example: yfs.disable.webpages.caching=N
yfs.rcp.pca.updates.dir		Set the property to specify the root folder under which the updates for the PCAs are located. It can be a shared folder on a network. Example: yfs.rcp.pca.updates.dir=<updates_directory>
yfs.rcp.pca.updates.cache.dir		Set following property to a local cache directory, where the updates need to be cached. IMPORTANT: The updates cache directory MUST NOT point to same location as the yfs.rcp.pca.updates.dir. The cache directory is a working directory and involves file I/O e.g. file creations and deletions, etc. Example: yfs.rcp.pca.updates.cache.dir=<updates_directory>
yfc.ui.ListPageSize	<number of records>	Set this to determine number of records shown on paginated list page. Example: yfc.ui.ListPageSize=30
# yfs.login.redirect.url		If the system needs to redirect to some custom url upon error in the login page, this property needs to be set to this url. If not specified, it will redirect to the default login page. This property can be specified in yfs.properties_ext file. Example: yfs.login.redirect.url=<REDIRECT_URL>
yfs.ui.MaxRecords	<number of records> Default = 200	This property set the number of records displayed on a list screen. Increase the application server JVM heap settings if these parameters are increased. Change will affect search limits for all users. Example: yfs.ui.MaxRecords=200
yfs.ui.queryTimeout	<number of seconds> Default = 60	This property sets the number of seconds Application Console will wait for a SQL Statement to execute. If the limit is exceeded, an SQLException is thrown. Set to zero means unlimited. Example: yfs.ui.queryTimeout=60

Property	Values	Description
yfs.ui.defaultEncoding	Default = UTF-8	This property controls the encoding of the HTML sent to the client. Example: <code>yfs.ui.defaultEncoding=UTF-8</code>
yfs.config.java.plugin.codebase		This property determines the JDK used by the configurator. If not passed, it defaults to <code>http://java.sun.com/update/1.5.0/jinstall-1_5_0_11-windows-i586.cab#Version=1,5,0,11</code>
yfs.config.password.noprompt	Valid values = Y, N Default = N	This enables the ability in the configurator or health monitor to popup a message asking the user to log back via the console. Example: <code>yfs.config.password.noprompt=N</code>
yfs.rcp.ui.pagesize	Default = 30	This property sets the maximum number of records to display on a single page in the Table on the Search List screen. This property is used by PCA Applications. Example: <code>yfs.rcp.ui.pagesize=30</code>
yfs.gridLayout.maximumRecords	Default is not set	This property is used to limit the number of records shown in grid screens in Mobile Application. For vt220 clients, because of memory implications, this property shouldn't be set to more than 7. If this property is not set or is not a positive integer, the default value would be taken as 7. Example: <code>yfs.gridLayout.maximumRecords=</code>
Warehouse Management		
yfs.solver.iterations.wavecreate		The number of solver iterations for Create Wave. Use this property to set the number of solver iterations through which the create wave Agent will undergo to optimize wave creation. Example: <code>yfs.solver.iterations.wavecreate=1</code>
yfs.cancelwave.offline.taskcount.greaterthan		Cancellation of wave happens in background depending upon the value of the property <code>yfs.cancelwave.offline.taskcount.greaterthan</code> . If the value of the property is for ex: 500 then cancellation of a wave, which has open tasks more than 500 included in it then the wave gets cancelled in offline mode. Waves that have open tasks equal to or less than 500 gets cancelled online. Example: <code>yfs.cancelwave.offline.taskcount.greaterthan=-1</code>

Property	Values	Description
yfs.cancelwave.offline.shipmentcount.greaterthan		<p>Cancellation of wave happens in background depending upon the value of the property <code>yfs.cancelwave.offline.shipmentcount.greaterthan</code>. If the value of the property is for ex: 500 then cancellation of a wave, which has shipments more than 500 included in it then the wave gets cancelled in offline mode. Waves that have shipments equal to or less than 500 gets cancelled online.</p> <p>Example: <code>yfs.cancelwave.offline.shipmentcount.greaterthan=-1</code></p>
yfs.cancelmove.offline.taskcount.greaterthan		<p>Cancellation of move request happens in background depending upon the value of the property <code>yfs.cancelmoverequest.offline.taskcount.greaterthan</code>. If the value of the property is for ex: 500 then cancellation of a move request, which has open tasks more than 500 included in it then the move request gets cancelled in offline mode. Move requests that have open tasks equal to or less than 500 gets cancelled online.</p> <p>Example: <code>yfs.cancelmove.offline.taskcount.greaterthan=-1</code></p>
yfs.containerization.maxshipmentsinoneround		<p>The containerizeWave API will use this property to select number of shipments to be considered for creating the outbound containers in one round. API will consider default value as 75 if value is not specified for this property.</p> <p>Example: <code>yfs.containerization.maxshipmentsinoneround=75</code></p>
com.yantra.wms.agent.server.WMSCancelWaveAgent.sleepTimeBeforeGetJob	Sleep time properties are in seconds	Set these properties required for cancel wave agent.
com.yantra.wms.agent.server.WMSCancelWaveAgent.sleepTimeBetweenTwoTries		Examples: <code>com.yantra.wms.agent.server.WMSCancelWaveAgent.sleepTimeBeforeGetJob=5</code> <code>com.yantra.wms.agent.server.WMSCancelWaveAgent.sleepTimeBetweenTwoTries=30</code>
com.yantra.wms.agent.server.WMSCancelWaveAgent.maxIterationWithNoProgress		<code>com.yantra.wms.agent.server.WMSCancelWaveAgent.maxIterationWithNoProgress=10</code>

Property	Values	Description
com.yantra.wms.agent.server.WMSCancelMoveAgent.sleepTimeBeforeGetJob	Sleep time properties are in seconds	Set these three properties required for cancel move agent. Examples: <code>com.yantra.wms.agent.server.WMSCancelMoveAgent.sleepTimeBeforeGetJob=5</code> <code>com.yantra.wms.agent.server.WMSCancelMoveAgent.sleepTimeBetweenTwoTries=30</code> <code>com.yantra.wms.agent.server.WMSCancelMoveAgent.maxIterationWithNoProgress=10</code>
com.yantra.wms.agent.server.WMSCancelMoveAgent.sleepTimeBetweenTwoTries		
com.yantra.wms.agent.server.WMSCancelMoveAgent.maxIterationWithNoProgress		
yfs.install.createWave.delayhours		The CREATE WAVE FOR SHIPMENT GROUP agent uses this property to increase the Available date (in hours) of YFS_Task_Q records with transaction key "CREATE_WAVE_TASK_Q.4001" against which no waves could be created. Agent considers default value as 1 if value is not specified for this property. If set to -1, the YFS_Task_Q records with transaction key "CREATE_WAVE_TASK_Q.4001" for which no waves are generated, will be deleted. Example: <code>yfs.install.createWave.delayhours=1</code>
yfs.createwave.dbRead.numShipmentLines	Default = 20000	The CREATE WAVE agent uses this property to read maximum number of shipment lines from database in a single query. Example: <code>yfs.createwave.dbRead.numShipmentLines=20000</code>
yfs.createwave.process.numShipmentLines	Default = 50000	The CREATE WAVE agent uses this property in order to determine the maximum number of shipment lines for optimal memory utilization during wave creation. Example: <code>yfs.createwave.process.numShipmentLines=50000</code>

Property	Values	Description
yfs.releaseWave.handleShortages	Valid values = <ul style="list-style-type: none"> RemoveShipmentsFromWave RemoveShipmentLinesFromWave RemoveShortageQuantitiesFromWave Default = RemoveShipmentsFromWave	<p>Based on this property, inventory shortages in shipments being releases in a wave will be handled Shipments thus created can be released in another wave. New shipments can be created for inventory having shortages.</p> <ul style="list-style-type: none"> Set it to RemoveShipmentsFromWave to remove shipments having inventory shortages from wave. Set it to RemoveShipmentLinesFromWave to create child shipments for shipment lines having shortages and let the shipment be released without these lines. Set it to RemoveShortageQuantitiesFromWave to create child shipments for shortage quantity and let the shipments be released with available inventory. <p>Example:</p> <pre>yfs.releaseWave.handleShortages=RemoveShipmentsFromWave</pre>
yfs.wms.resourceplanning.QueueName		Set these JMS properties to enable resource planning move.
yfs.wms.resourceplanning.ProviderURL		Examples:
yfs.wms.resourceplanning.QCFLookUp		<pre>yfs.wms.resourceplanning.QueueName=DefaultAgentQueue</pre> <pre>yfs.wms.resourceplanning.ProviderURL=t3://localhost:7001</pre> <pre>yfs.wms.resourceplanning.QCFLookUp=AGENT_QCF</pre>
yfs.wms.zonetransattr.refreshrate	Default = 600 seconds	<p>getZoneDetails API uses the below mentioned property to refresh the zone attributes: available volume, available weight, pending volume and pending weight in the table YFS_TRAN_ZONE_ATTRS. These zone attributes are aggregates of the location attributes. Zone attributes # are refreshed after specified time intervals which is defined by this property.</p> <p>Example:</p> <pre>yfs.wms.zonetransattr.refreshrate=600</pre>

Property	Values	Description
yfs.releaseWave.pickalgorithm		releaseWave uses the below mentioned property to select the algorithm to be used to determine pick locations. This property is valid only when split constraints are defined. ConstraintSolver: Constraint Solver Algorithm is used. The default value of this property is set to blank. When set to blank, a simple iterative algorithm is used. Example: <code>yfs.releaseWave.pickalgorithm=</code>
yfs.releaseWave.SortShipmentsByDateAndKey	Valid values = Y/N Default = N	During Release wave, if shipments need to be allocated quantity based on requested shipment dates, set this property to "Y". Example: <code>yfs.releaseWave.SortShipmentsByDateAndKey=N</code>
yfs.default.inventorystatus.for.newinventory		In case of counting through mobile application if user counts new inventory, system currently displays first inventory status of the node as defaulted. If this behavior needs to be overridden, then the below property needs to be set with valid inventory status which will be used as default inventory status for new inventory. If this property is not set, system will retain the current behavior. Example: <code>yfs.default.inventorystatus.for.newinventory =</code>
yfs.closemanifest.online	Valid values = Y/N	Manifesting options. Set this property to N to close manifest asynchronously using the CLOSE_MANIFEST Agent. Example: <code>yfs.closemanifest.online=Y</code>
yfs.confirmbatch.online	Valid values = Y/N	Batch completion options. Set this property to N to complete batch asynchronously using the REQ_BATCH_COMPLETION Agent. Example: <code>yfs.confirmbatch.online=Y</code>
yfs.reopentask.time	Default = 1 hour	The re-open task agent will change the status of a suggested task back to open status, if the task is not modified for the time specified by this property. This property is specified in hours. Example: <code>yfs.reopentask.time=1</code>
yfs.serial.receiving	Valid values = Y/N	Set this property to 'Y' to process one serial at a time during receiving i.e on scanning one serial user is taken to Disposition Code entry screen. Example: <code>yfs.serial.receiving=N</code>

Property	Values	Description
yfs.releasetaskagent.mode	Valid values = 01: Location Driven. All the pending tasks from the location are evaluated. 02: Reference(eg. ShipmentNo) Driven. All the pending tasks for the reference are evaluated. Default = 01	releaseTaskAgent uses this property to select the mode of execution for releasing tasks. Example: <code>yfs.releasetaskagent.mode=01</code>
yfs.retain.pack.tasks.for.minute	Default = 60 minutes	In case of item driven packing, shipment level pack tasks are created in order to reserve the shipment against the packer. Once the shipment packing is complete, these tasks are deleted from DB. If for some reason packing for this shipment cannot be completed by the packer, these idle pack tasks block other users from packing this shipment. This property can be used to define maximum time, in minutes, for which the shipment level idle pack tasks are retained. If it is set to 120, and a user scans a Pallet in Pack HSDE, pack tasks created 120 minutes before will be deleted so that the unpacked blocked shipment becomes available to other users. Example: <code>yfs.retain.pack.tasks.for.minutes=60</code>
yfs.override.user.constraints.for.manually.assigned.task	Valid values = Y/N Default = N	Based on this property, the maximum user constraints defined for a task type will be evaluated. If task is manually assigned to a user and this property is set to Y, the maximum user constraint defined for task type will not be honoured. To retain the existing behavior, set this property to N. Example: <code>yfs.override.user.constraints.for.manually.assigned.task=N</code>
yfs.displayTaskInfoDetails	Valid values = Y/N Default = N	Set this property to 'Y' to show the task details (Pick Location, Item ID and Quantity) in RF mobile terminal when tasktype is defined to say pick onto equipment. This provides user an ability to determine whether he needs to carry a pallet/case to pick inventory even before he goes to the pick location. Example: <code>yfs.displayTaskInfoDetails=N</code>

Property	Values	Description
yfs.allow.container.quantity.adjustment	Valid values = Y/N	In case of picking through mobile application if user picks a full LPN, system currently doesn't provide an option to adjust the discrepancies. To allow user to adjust the discrepancies in the license plates, the property needs to be set to Y. If this property is not set or set to N, system will retain the current behavior. Example: yfs.allow.container.quantity.adjustment=

A.4 dbclassCache.properties.*

The dbclassCache.properties file contains a list of the Sterling Multi-Channel Fulfillment Solution tables and the associated DBCache class. Every table listed will be registered with the cache manager with the default cache sizes. The dbclassCache.properties file also contains the global settings for the DBCache classes, and additional properties that can be uncommented and set as needed. Do not make changes directly to the dbclassCache.properties file; instead, use the customer_overrides.properties file. See [Chapter 12, "Configuring Properties"](#) for information about using customer_overrides.properties.

Property	Values	Description
sci.globalcache.select.size	Any positive integer, defaults to 10000	Default maximum size of the "select" database cache for a given table. This is the number of select statements that return a single record that will be cached.
sci.globalcache.list.size	Any positive integer, defaults to 10000	Default maximum size of the "list" database cache for a given table. This is the number of select statements that return a multiple records that will be cached.
sci.globalcache.count.size	Any positive integer, defaults to 10000	Default maximum size of the "count" database cache for a given table. This is the number of select statements that count records that will be cached.
sci.globalcache.object.size	Any positive integer, defaults to 10000	Default maximum size of the "object" database cache for a given table. This is the total number of resulting database objects across the "select" and "list" caches that will be cached. If this limit is reached, all database caches for the table are dropped

The dbclassCache.properties file also lists the cacheable tables, in the form of <TableName>.class=<cache implementation>. This would only be changed/added by applications.

Each table also supports overriding the defaults provided in the globalcache settings. For example, using YFS_COMMON_CODE as the table name:

Property	Values	Description
YFS_COMMON_CODE.enabled	true/false. Default value is true.	Whether the database cache is enabled. Set to false to disable caching.
YFS_COMMON_CODE.select.size	Any positive integer, defaults to value of sci.globalcache.select.size	Override the “select” cache size for the given table.
YFS_COMMON_CODE.list.size	Any positive integer, defaults to value of sci.globalcache.list.size	Override the “list” cache size for the given table.
YFS_COMMON_CODE.count.size	Any positive integer, defaults to value of sci.globalcache.count.size	Override the “count” cache size for the given table.
YFS_COMMON_CODE.object.size	Any positive integer, defaults to value of sci.globalcache.object.size	Override the “object” cache size for the given table.

A.5 Sandbox.cfg

Sandbox.cfg contains properties related to system and database information.

Note: The `sandbox.cfg` file is not used at runtime by the product. If you change a parameter in the `sandbox.cfg` file at any time, you must run the `setupfiles` script so that the runtime property files are re-created with the updated values.

Property	Description
Agent	
AGENT_JAVA_HOME	Overrides the java that the agents use. For instance, if, with WebSphere, you wanted to use the IBM jdk, you could set this parameter to a new value.
Internal	

Property	Description
DB_DATA	Only used at install time. Database name to connect to. Note: To change this database property after installation, use the database specific properties (see the Database section).
DB_DRIVERS	Only used at install time. Full path to JDBC driver file. Note: To change this database property after installation, use the database specific properties (see the Database section).
DB_HOST	Only used at install time. Database host to connect to. Note: To change this database property after installation, use the database specific properties (see the Database section).
DB_JAR_DIR	Only used at install time. Note: To change this database property after installation, use the database specific properties (see the Database section).
DB_PASS	Database password to connect with.
DB_POOL	Note: To change this database property after installation, use the database specific properties (see the Database section).
DB_PORT	Only used at install time. Database listener port. Note: To change this database property after installation, use the database specific properties (see the Database section).
DB_USER	Database login ID to connect with. Note: To change this database property after installation, use the database specific properties (see the Database section).
DB_VENDOR	Required. The DB vendor to use. Valid values are Oracle, DB2, MSSQL. Note: To change this database property after installation, use the database specific properties (see the Database section).
INSTALL_DIR	Required. Directory in which to install.
JDBC_DRIVER	Path to database driver file.
SI_LICENSE_AVAILABLE	Indicator of if a license is being passed in and is required for installation. Valid values are Yes and No (default).

Property	Description
NOAPP_HOME	These properties should only be changed as a group.
CLASS_DIR	
HOME_DIR	
VENDORS_DIR	
DEPLOYED_APP_DIR	
BIN_DIR	
APPBEANS_DIR	
SVC_DIR	
DIST_DIR	
Database	These are database-related properties that can be changed after installation. Use the vendor-specific properties in this section to make changes to your installed database's properties.
	Note: You cannot change the database provider for a specific SSCAP instance after installation.
DB_SCHEMA_OWNER	Default schema/schema-owner for the provided login ID.
JDBC_VENDOR	JDBC driver vendor. Used when multiple vendor are available. Default is Microsoft.
DB2_DATA	If using DB2, the database name to connect with.
DB2_HOST	If using DB2, the database host to connect to.
DB2_PASS	If using DB2, the database password to connect with.
DB2_PORT	If using DB2, the database listener port.
DB2_USER	If using DB2, the database login ID to connect with.
MSSQL_DATA	If using MSSQL, the database name to connect with. Valid for both MSSQL2000 and MSSQL2005.
MSSQL_HOST	If using MSSQL, the database host to connect to. Valid for both MSSQL2000 and MSSQL2005.
MSSQL_PASS	If using MSSQL, the database password to connect with. Valid for both MSSQL2000 and MSSQL2005.

Property	Description
MSSQL_PORT	If using MSSQL, the database listener port. Valid for both MSSQL2000 and MSSQL2005.
MSSQL_USER	If using MSSQL, the database login ID to connect with. Valid for both MSSQL2000 and MSSQL2005.
ORA_HOST	If using Oracle, the database host to connect to.
ORA_PASS	If using Oracle, the database password to connect with.
ORA_PORT	If using Oracle, the database listener port.
ORA_USER	If using Oracle, the database login ID to connect with.

Property	Description
DB Pooling	
MIN_TRANS_POOL	mssqlPool.initsize
MAX_TRANS_POOL	mssqlPool.maxsize
Implementation	These are properties that can be changed after installation.
LOAD_FACTORY_SETUP	Indicates whether or not you want to load factory setup defaults during installation or manually after installation.
AUDIT_LOAD_DEFAULTS	Valid values are true/false. When set to true the audits are generated when the loadDefaults script is run.
LOG_DIR	Use to override the logging directory. For example, if you want to deploy the ear on another server you could set the parameter to a new value, run setupfiles, then build the ear. The application would then log to the directory you mentioned.
NO_DBVERIFY	Valid values are true/false. When set to true during installation and installservice, dbverify will not be run. This means that the Sterling Multi-Channel Fulfillment Solution will not generate DDL to make the database like the XML entity repository.
REINIT_DB	Whether the Sterling Multi-Channel Fulfillment Solution should initialize the database or not. Valid values are Yes (default) and No.

A

- access control, 179, 189
- Activesync, 132
- agent criteria
 - override, 171
- authentication
 - using JAAS, 19
 - using LDAP, 19

B

- barcode printing
 - WebLogic, 179
 - WebSphere, 189
- browser
 - clearing cache, 203

C

- capacity planning, 41, 68
- catalog name for database, 92, 112
- clearing browser, 203
- clearing java plugin caches, 203
- client character, 203
- client characters
 - displaying special characters, 203
 - displaying unicode character, 203
- cold boot
 - reinstallation, 135
- component object model., 25
- COM+ (extended component object model), 25
- Configuring

- server images, 223
- configuring application server, 34
 - JDK upgrades, 34
- copying WMS label formats, 127
- cross-site script vulnerabilities
 - malicious HTML, 152
 - prevention, 152
 - enabling html encoding, 152
- custom classes deployment, 183, 193, 200

D

- data migrator, 167
 - log files, 169
 - errors, warnings, 169
 - properties file, 149
 - restart file, 169
 - task definition files, 167
- Database catalog name, 92, 112
- database name, 92, 112
- database schema creation, 160
- database security, 27
 - credit card encryption, 28
- database sizing, 41, 68
 - capacity planning, 41, 68
 - disk estimation
 - DOM module, 42, 69
 - nWMS module, 45, 72
 - future disk estimation, 51, 78
- database user privileges, 53, 83
 - DB2, 60
 - Oracle, 53, 83
 - oracle

- administrative user, 53, 84
 - application user, 54, 84
- SQL server, 79
- database verification, 160
- databases
 - configuring, 67
 - DB2 8.1 Fixpack 5, 63, 64
 - Oracle, 58, 59, 88, 89
 - Oracle (UNIX/Linux), 96
 - Oracle (Windows), 115
 - reinstalling, 68
- DB2
 - codepage selection, 60
 - configuring, 60
 - installation, 60
- DB2 database
 - production environment, 60
 - sizing, *See Also* database sizing, 55, 60, 85
- DB2 8.1 Fixpack 5 databases, 63, 64
- deploying
 - custom classes (WebLogic), 183, 193, 200
- deploying Sterling Multi-Channel Fulfillment Solution, 175
- deployment architecture
 - analyzing
 - authentication mechanism, 19
 - current security infrastructure, 18
 - data encryption, 19
 - network topology, 20
- disk estimation
 - DOM module, 42, 69
 - before estimating, 43, 44, 69, 71
 - estimation methodology, 42, 69
 - steps involved, 44, 71
 - future requirements, 51, 78
 - nWMS and DOM module
 - steps involved, 47, 74
 - nWMS module, 45, 72
 - before estimating, 46, 47, 73, 74
 - estimation methodology, 45, 72
 - nWMS module only
 - steps involved, 49, 76
- DOM module
 - disk space estimation, 44, 71
- DOM module disk estimation, 42, 69

E

- EFrame_indexadds.sql, 162
- EFrame_indexdrops.sql, 162
- EFrame_sequence.sql, 161
- EFrame_TableChanges.sql, 86, 161
- EFrame_tablechanges.sql, 162
- EFrame_tabledrops.sql, 162
- EFrame_TextIndexAdds.sql, 162
- EFrame_TextIndexDrops.sql, 162
- EFrame_TextIndexModify.sql, 162
- EFrame_TextIndexUpdates.sql, 162
- EFrame_UpdateQueries.sql, 162
- EJB (Enterprise JavaBeans), 21
- ejbdeploy.sh, 186
- enterprise javabeans. *See* EJB, 21
- environment variable
 - INSTALL_DIR, xxv
 - INSTALL_DIR_OLD, xxv
- estimates of database, 41, 68

F

- factory defaults, 165
 - installer restart file, 165

G

- generating EJB stubs, 186
- graph display on WebLogic, 179
- graph display on WebSphere, 189

H

- history database, 43, 46, 70, 73
- HP-UX
 - ncurses installation, 139
- HTML characters
 - impacting display, 152
- HTTP
 - in-memory session replication, 179
- HTTP (hypertext transfer protocol), 22
- HTTPS (hypertext transfer protocol secure), 24
- hypertext transfer protocol secure. *See* HTTPS, 24

hypertext transfer protocol. See HTTP, 22

I

- installation checklist, 1
- installation parameters
 - database catalog name, 92, 112
- Installing, 34
- installing
 - DB2
 - JDK, 114
 - JDK (UNIX/Linux), 96
 - Linux, 98
 - Oracle
 - SQL Server
 - UNIX, 98
 - Windows, 118, 119
- installing application server, 33
- installing database schema, 160
- installing database software
 - DB2. See DB2
 - Oracle. See Oracle
 - SQL Server. See SQL Server
- installing database software (Windows), 41, 67
- installing DB2, 60
- installing jasperreports, 127
- installing mobile application, 131
- installing Oracle, 51, 82
- installing print server, 125
- installing SQL Server, 78
- installing weighing scale, 129
- installing (from command line)
 - Linux, 99, 104
 - UNIX, 99, 104
- installing (from silent install file)
 - Linux, 108
- installing (from Windows)
 - Linux, 122
 - UNIX, 108, 122
- INSTALL_DIR, xxv
- INSTALL_DIR_OLD, xxv
- Internet Explorer security
 - adding trusted website, 31
 - browser settings, 28
- invoking

custom classes, 183, 193, 200

J

- JAAS (Java Authentication and Authorization Service), 19
- jasperreports
 - installation, 127
- Java Authentication and Authorization Service. See JAAS, 19
- java messaging service. See JMS, 22
- java naming and directory interface. See JNDI, 23
- java plugin
 - clearing cache, 204
- java protocol
 - disabling, 21
 - EJB, 21
 - HTTP, 22
 - JMS, 22
- java protocols
 - securing, 23
 - COM+, 25
 - EJB, 23
 - HTTP API tester, 24
- Java 2 Software Development Kit. See JDK
- JavaServer pages. See JSP, 23
- JBoss
 - using Web services, 198
- jcrt.jar, 178
- JDK
 - installing, 114
 - installing (UNIX/Linux), 96
- JMS (java messaging service), 22
- JNDI (java naming and directory interface), 23
- jnet.jar, 178
- JSP (JavaServer Pages), 23
- jsse.jar, 178
- JVM settings
 - for WebSphere, 187

L

- LAN (local area network), 21
- LDAP user authentication, 149
 - assumption, 149

- for application consoles, 149
- modifying configurator UI, 151
- modifying weblogic startup file, 151
- modifying yfs.properties
 - customer_overrides.properties, 151
- password expiration, 150
 - change password link, 150
 - expire in days, 150
- properties, 150
- LDAP (lightweight directory access protocol), 19
- lightweight directory access protocol. See LDAP, 19
- Linux
 - installing, 98
 - installing (from command line), 99, 104
 - installing (from silent install file), 108
 - installing (from Windows), 122
 - silent installation file, 97, 116
 - silent installation file parameters, 97
 - UNIX accounts, 96
- LLM (software label manager), 125
- loading factory defaults, database factory
 - defaults, 165
- local area network. See LAN, 21
- software label manager, 125
 - bar code label, 125
 - defining printers, 126
 - designing label, 125
 - for UNIX systems, 126
 - supported modes, 125
- log files
 - best practices, security, 25
 - for data migrator, 169
 - setting up, 152
- logging
 - properties, 153
 - utility, 153
- log4j configuration file, 154
- log4j utility, 153
- LPM (software print server), 125

M

- maximum performance, 3
- Microsoft .NET, 136

- Microsoft SQL Server. See SQL Server
- mobile application
 - Activesync for copying files, 132
 - installation, 131
- mobile installation
 - cold boot reinstallation, 135
 - installing libiconv
 - HP-UX Itanium, 139
 - installing ncurses, 139
 - steps to install, 132
- mobile terminals
 - device requirements
 - barcode scanner, 11
 - keys, 11
 - minimum memory, 12
 - network connectivity, 12
 - screen resolution, 12
 - synchronization, 11

N

- named pipes, 79
- namedwebservices.xml, 182, 199
- ncurses, 139
- network topology
 - deployment over internet, 20
 - deployment over LAN, 21
 - deployment over VPN, 20
- networked WMS module disk estimation, 45, 72
- nWMS and DOM module
 - disk space estimation, 47, 74
- nWMS module only
 - disk space estimation, 49, 76

O

- operating system permissions, 26
- Oracle
 - configuring, 55, 85
 - copying JDBC driver, 53, 83
 - installation, 51, 82
 - running create instance, 52, 82
- Oracle database
 - production environment, 55, 85
 - creating views, 55, 85

- setting connection properties, 234
- Oracle databases, 58, 59, 88, 89
 - UNIX/Linux, 96
 - Windows, 115

P

- password expiration, 150
- Pocket PC mobile terminal
 - reinstalling on cold boot, 135
- PocketPC mobile terminal
 - installing Sterling, 131
- post installation recommendations, 25
- pre-compiling JSP files, 176
- Preparing the Database Server, 41, 67
- print server
 - configuration utility, 126
 - installing software components, 125
- print server installation, 125
- procuring storage, 44, 47, 71, 74
- properties file
 - for data migrator, 149

R

- recommended data encryption
 - SSL 128-bit, 19
 - VPN 3DES, 19
- reinstalling
 - databases, 68
- remote method invocation. See RMI, 24
- restart file
 - for data migrator, 169
- restart files
 - preserving, 169
- retention policy, 43, 70
- RMI (remote method invocation), 24
- running
 - Oracle scripts, 56, 61, 86
 - SQL Server scripts, 79

S

- security

- database security, 27
- deployment architecture, 18
- Internet Explorer settings, 28
- java protocol See java protocol
- web security See web security
- security plan
 - creating, 17
- session replication, 179
 - including session descriptors, 180
- Setting Oracle Database Connection Properties, 234
- setting up
 - dbverify utility, 160
 - Oracle scripts, 55, 85
 - Web services, 181, 192, 198
 - WebLogic scripts, 176
- setting up Sterling Multi-Channel Fulfillment Solution, 143
- SID, 92, 112
- silent installation file
 - Linux, 97, 116
 - parameters for Linux, 97
 - parameters for UNIX, 97
 - parameters for Windows, 116
 - UNIX, 97, 116
- SQL Server
 - configuring, 79
- SQL Server database
 - enabling named pipes, 79
 - enabling TCP/IP, 79
 - production environment, 79
 - running scripts, 79
 - sizing, See Also database sizing, 79
- stack traces
 - eliminating, 178
- Sterling Multi-Channel Fulfillment Solution
 - deploying, 175
- Sterling Multi-Channel Fulfillment Solution Mobile Application
 - installation
 - PocketPC mobile terminal, 131
- Sterling Multi-Channel Fulfillment Solution utilities
 - db verify, 158
 - installation, 157
 - db verify, 158

- installService, 158
- install3rdParty, 158
- loadDefaults, 157
- setupfiles, 158
- upgrade

- MigrationValidator, 167
 - migrator, 167

- upgrade, 167

- System Console, 189

- system requirements

- minimum, 3

- UNIX/Linux, 96

- Windows, 114

T

- task definition file

- childtask element, 169

- task element, 168

- task definition files

- for data migrator, 167

- task definition file

- taskinfo element, 168

- TCP/IP, 79

- time-triggered transactions

- override agent criteria, 171

- transaction data

- truncating, 170

- transactional database, 43, 46, 70, 73

- truncating transaction data, 170

- types of order, 44, 71

U

- UNIX

- accounts, 96

- installing, 98

- installing (from command line), 99, 104

- installing (from silent install file), 108

- installing (from Windows), 122

- silent installation file, 97, 116

- silent installation file parameters, 97

- UNIX accounts

- Linux, 96

- UNIX, 96

- Utilities

- development, 169

- transaction data truncation tool, 170

- runtime, 170

- agent server, 171

- agent trigger, 171

- Integration server, 170

- setting up, 171

V

- verifying database, 160

- extra objects, 163

- virtual private network. See VPN, 20

- VPN (virtual private network), 20

- VT220 Mobile Terminal, 131, 138

- installing, 138

- steps to install, 139

W

- web security

- routing, 27

- session security, 26

- web server executables, 27

- writing log files, 25

- Web services, 181, 192, 198

- WebLogic

- creating EAR

- file descriptors, 183

- steps involved, 183

- eliminating stack traces, 178

- setting client character display, 203

- setting HTTP in-memory session

- replication, 179

- using Web services, 181, 192

- WebLogic application server

- setting up, 176, 185

- weblogic.xml, 180

- WebSphere

- avoiding warning messages, 185

- configuring JVM settings, 187

- creating EAR

- steps involved, 193

- invoking EJBs, 186
- making EJB calls, 186
- specifying memory parameters, 185
- weighing scale
 - installation, 129
 - steps to install, 129
- Win CE 5.0 mobile terminal
 - reinstalling on cold boot, 137
- WinCE mobile terminal, 131
 - installing Sterling, 131
- WinCE mobile terminals
 - reinstalling on cold boot, 136
- Windows
 - installing, 118, 119
 - silent installation file parameters, 116
- WMS installation
 - defining printers, 126

Y

- yantraejb.jar, 180, 187, 191, 198
- yantra.jar, 197
- yantra.war, 180, 191, 198
- yantrawebsservices.war, 180, 191, 198
- yantrawsbe.jar, 180, 191, 198

