

Upgrading webMethods and Intelligent Business Operations Products

Version 9.8

April 2015

This document applies to webMethods Product Suite Version 9.8 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2011-2015 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Table of Contents

About this Guide.....	9
Document Conventions.....	9
Online Information.....	10
Upgrades Covered in This Guide.....	11
Upgrades Covered in This Guide.....	12
Release Numbering Exceptions.....	13
Guides Needed to Perform Upgrades.....	13
Critical Requirements For Successful Upgrade.....	15
Critical Upgrade Requirements.....	16
Product Requirements.....	16
API-Portal Requirements.....	16
9.6 or 9.7 Upgrade: webMethods Broker Requirements.....	17
CentraSite Requirements.....	17
Integration Server Clustering Requirements.....	17
Mobile Administrator Requirements.....	18
My webMethods Server Requirements.....	18
Upgrading Products to the New Release.....	19
Read Critical Requirements for Successful Upgrade.....	20
Migrating Across Different Hardware or Operating Systems.....	20
Install the 9.8 Products.....	20
Prepare the Old Environment for Upgrade.....	22
Prepare Databases.....	22
Install Fixes on Products.....	22
Prepare the Old API-Portal.....	22
Prepare the Old CentraSite.....	23
Prepare the Old Software AG Designer.....	23
Export Integration Server Definitions.....	23
Export CloudStreams Server Definitions and Projects.....	24
Export Event Bus Console Configurations.....	24
Export Preferences.....	24
Prepare the Old Integration Server.....	24
Prepare the Old OneData.....	26
Prepare the Old Optimize.....	26
Shut Down the Old Products.....	27
Migrate Database Components.....	27
Back Up the RDBMSs.....	27
Upgrade the RDBMSs.....	28
Shut Down Products that Connect to Database Components.....	28
Before You Migrate Trading Networks Database Components.....	28

Special Considerations for Migrating Other Database Components.....	28
Install Fixes on Database Scripts.....	29
Database Component Names.....	29
Migrate Database Components.....	30
Complete the Side-by-Side Upgrade.....	30
Migration Utilities.....	31
Use of Zip File as Source of Data to Migrate.....	31
Create a Zip File for My webMethods Server.....	31
Create a Zip File for All Other Products.....	32
Complete the webMethods and IBO Infrastructure Upgrade.....	35
Overview.....	36
Before Migrating Data.....	36
Run the Infrastructure Migration Utility.....	36
9.0, 9.5, 9.6, or 9.7 Upgrade: Update the Software AG Runtime NERV Configuration.....	37
Complete Final Upgrade Tasks.....	37
Complete the EDA/NERV, Terracotta, and Messaging Upgrades.....	39
Complete the EDA/NERV Upgrade.....	40
Run the EDA/NERV Migration Utility.....	40
Update Paths.....	40
Complete Final Upgrade Tasks.....	40
Complete the Terracotta Upgrade.....	40
Complete the Universal Messaging Upgrade.....	41
Before Migrating Data.....	41
Run the Universal Messaging Migration Utility.....	41
Perform a Custom Migration.....	41
Migrate Using Imported Settings.....	42
Migrating Using Imported Settings with Prompting.....	42
Migrate Using Imported Settings without Prompting.....	42
Migrate Custom Changes, Data Directory, and License File.....	43
Complete Final Upgrade Tasks.....	43
Complete the webMethods Broker Upgrade.....	43
Overview.....	43
Create Broker Server 9.6 and Use the Old Storage.....	44
Migrate on Multi-Home Machines.....	45
Complete Final Upgrade Tasks.....	45
Completing the My webMethods Server Upgrade.....	47
Before Migrating Data.....	48
Run the My webMethods Server Migration Utility.....	48
Perform a Custom Migration.....	48
Migrate Using Imported Settings.....	50
Migrate Using Imported Settings with Prompting.....	50
Migrate Using Imported Settings without Prompting.....	50
8.2 Upgrade: Migrate the My webMethods Server Infrastructure.....	50

9.0, 9.5, 9.6, or 9.7 Upgrade: Migrate the My webMethods Server Infrastructure.....	51
Initialize My webMethods Server.....	51
Update Host Names.....	51
Complete Final Upgrade Tasks.....	52
Completing the Integration Server and Hosted Products Upgrade.....	53
Before Migrating Data.....	54
Run the Integration Server Migration Utility.....	54
Perform a Custom Migration.....	54
Migrate Using Imported Settings.....	57
Migrate Using Imported Settings with Prompting.....	57
Migrate Using Imported Settings without Prompting.....	58
Other Actions Performed by the Migration Utility.....	58
Complete the Integration Server Upgrade.....	59
Migrate the Flat File Package Properties File.....	59
Customize Integration Server Startup Files.....	59
Update WSDLs.....	59
Connect Integration Server Functions to Database Components.....	60
Update Host Names.....	60
9.0, 9.5, 9.6, or 9.7 Upgrade: Update the Integration Server NERV Configuration.....	60
Complete Final Upgrade Tasks.....	61
Complete the ActiveTransfer Upgrade.....	61
Complete the CloudStreams Upgrade.....	61
Complete the Trading Networks Upgrade.....	62
Complete Adapter Upgrades.....	63
webMethods Adapter for JDBC.....	63
webMethods Package for Microsoft .NET.....	63
webMethods PeopleSoft EnterpriseOne Adapter.....	63
webMethods SAP Adapter.....	63
Complete eStandards Module Upgrades.....	64
webMethods FIX Module.....	64
webMethods SWIFTNet Module.....	64
Completing the Software AG Designer, Business Process Run-time, and Blaze Upgrades.....	65
Migrate Software AG Designer Data.....	66
Before Migrating Data.....	66
Migrate Business Process and Business Rule Projects.....	66
Migrate Task Application Projects.....	67
Migrate CloudStreams Server Definitions and Projects.....	67
Migrate Integration Server Definitions.....	68
Complete the Business Process Run-time Upgrade.....	68
Update Host Names.....	69
Migrate Closed Loop Analytics.....	69
Complete Final Upgrade Tasks.....	69
Complete the Blaze Rules Upgrade.....	69

Completing the Optimize Upgrade.....	71
Configure and Start the Terracotta Server Array.....	72
Update Connection to Other Products.....	72
Update Connection to Process Engine.....	72
Update Connection to Optimize Support Package.....	72
Complete the Infrastructure Data Collector Upgrade.....	73
8.2 Upgrade: Restart Monitoring of Clustered Integration Servers.....	73
9.7 Upgrade: Start Monitoring 4.1 or Later Terracotta Server Arrays.....	73
Migrate Customizations to the Java Service Wrapper.....	73
Run the Infrastructure Data Collector Migration Utility.....	73
Before Migrating Data.....	74
Perform a Custom Migration.....	74
Migrate Using Imported Settings.....	74
Migrate Using Imported Settings with Prompting.....	75
Migrate Using Imported Settings without Prompting.....	75
Deploy the Optimize Environment.....	75
Not Reuse the Old Optimize CCS Environment Definition.....	75
Reuse the Old Optimize CCS Environment Definition.....	76
Deploy the Optimize 9.8 Environment.....	77
Switch from webMethods Broker to Universal Messaging.....	77
Complete Final Upgrade Tasks.....	78
Completing the Asset Build Environment and Deployer Upgrades.....	79
Complete the Asset Build Environment Upgrade.....	80
Complete the Deployer Upgrade.....	80
Before Migrating Deployer Settings, Server Aliases, Target Groups, and Projects.....	80
Migrate All Deployer Settings, Server Aliases, Target Groups, and Projects Using the Integration Server Migration Utility.....	81
Migrate Deployer Settings, Server Aliases, Target Groups, and Projects Using the Deployer Migration Feature.....	81
Start the Old Deployer and Deployer 9.8 and Migrate Global (Default) Settings.....	81
Migrate Server Aliases (Connection Information).....	82
Migrate Target Groups.....	82
Migrate Deployment Projects.....	83
Complete Final Upgrade Tasks.....	83
Completing the Command Central Upgrade.....	85
Before Migrating Data.....	86
Migrate the Command Central Infrastructure.....	86
Run the Command Central Migration Utility.....	87
Perform a Custom Migration.....	88
Migrate Using Imported Settings.....	88
Migrate Using Imported Settings with Prompting.....	89
Migrate Using Imported Settings without Prompting.....	89
Complete the Command Central Upgrade.....	89

9.6 or 9.7 Upgrade: Restore Trusted Authentication for JAAS.....	89
Update the Command Central NERV Configuration.....	89
Complete Final Upgrade Tasks.....	89
Complete the Platform Manager Upgrade.....	91
9.0 or 9.5 Upgrade: Complete the Platform Manager Upgrade.....	92
Before Migrating Data.....	92
Migrate the Platform Manager Infrastructure.....	92
Update the Platform Manager NERV Configuration.....	94
Complete Final Upgrade Tasks.....	94
9.6 or 9.7 Upgrade: Complete the Platform Manager Upgrade.....	94
Before Migrating Data.....	94
Migrate the Platform Manager Infrastructure.....	94
Run the Platform Manager Migration Utility.....	95
Perform a Custom Migration.....	95
Migrate Using Imported Settings.....	96
Migrate Using Imported Settings with Prompting.....	96
Migrate Using Imported Settings Without Prompting.....	96
Complete the Platform Manager Upgrade.....	97
Restore Trusted Authentication for JAAS.....	97
Update the Platform Manager NERV Configuration.....	97
Complete Final Upgrade Tasks.....	97
Completing the API-Portal and CentraSite Upgrade.....	99
Completing the API-Portal Upgrade.....	100
Completing the CentraSite Upgrade.....	101
Before Migrating Data.....	101
Migrate the Data.....	101
Complete the Upgrade.....	103
Complete the Mobile Upgrade.....	105
9.7 Upgrade: Complete the Mobile Designer Upgrade.....	106
Complete the Mobile Administrator Upgrade.....	106
Overview.....	106
Run the Updater and Verify Configurations.....	107
Update Build Nodes.....	107
Update the App Store Client.....	107
Complete Final Upgrade Tasks.....	108
Complete the OneData Upgrade.....	109
Run the OneData Migration Utility.....	110
Migrate Customizations to the Java Service Wrapper.....	110
8.4 Upgrade: Enable New Logging Method.....	110
Complete Final Upgrade Tasks.....	110
Completing the Presto Upgrade.....	111
Overview.....	112

Set Up the Presto Repository.....	112
Set Up Repository When Using the Embedded Derby Database.....	112
Set Up Repository When Using Any Other Database.....	112
Update the Presto Configuration and Copy Extensions.....	113
Update When an External Configuration Directory Was Used.....	113
Update When No External Configuration Directory Was Used.....	114
3.7 or 3.8 Upgrade: Upgrade the Integrated MashZone Server and RTBS Server Configurations, and MashZone Feeds.....	114
Execute Upgrade Commands.....	114
3.6 or 3.7 Upgrade: Update Mashups that Use RAQL Queries.....	115
Update the Snapshots Repository Connection Information.....	115
Complete Final Upgrade Tasks.....	116
Completing Final Upgrade Tasks for All Products, and Troubleshooting.....	117
Complete Final Upgrade Tasks for All Products.....	118
Troubleshooting.....	119

About this Guide

This guide explains how to upgrade webMethods and Intelligent Business Operations products from 8.2, 9.0, 9.5, 9.6, 9.7 to 9.8.

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies storage locations for services on webMethods Integration Server, using the convention <i>folder.subfolder:service</i> .
UPPERCASE	Identifies keyboard keys. Keys you must press simultaneously are joined with a plus sign (+).
<i>Italic</i>	Identifies variables for which you must supply values specific to your own situation or environment. Identifies new terms the first time they occur in the text.
Monospace font	Identifies text you must type or messages displayed by the system.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the symbol.
[]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at <http://documentation.softwareag.com>. The site requires Empower credentials. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

You can find product information on the Software AG Empower Product Support website at <https://empower.softwareag.com>.

To submit feature/enhancement requests, get information about product availability, and download products, go to [Products](#).

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the [Knowledge Center](#).

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at <http://techcommunity.softwareag.com>. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

1 Upgrades Covered in This Guide

■ Upgrades Covered in This Guide	12
■ Release Numbering Exceptions	13
■ Guides Needed to Perform Upgrades	13

Upgrades Covered in This Guide

This guide explains how to upgrade webMethods and Intelligent Business Operations products from release 8.2, 9.0, 9.5, 9.6, or 9.7 to release 9.8 using a side-by-side procedure.

The upgrades listed below are not covered in this guide; instead, see the indicated information source.

Upgrade	Information
From a release prior to 8.2	See <i>Supported Upgrade Paths</i>
ApplinX	See ApplinX documentation
webMethods Broker to Universal Messaging	See <i>Migrating from webMethods Broker to webMethods Universal Messaging</i>
Content Service Platform	Contact Software AGGlobal Consulting Services
MashZone or Process Performance Manager	See MashZone or Process Performance Manager documentation
Mobile Administrator on a virtual machine (This guide explains how to upgrade Mobile Administrator on a physical machine)	Uninstall Mobile Administrator from the virtual machine (see <i>Installing webMethods and Intelligent Business Operations Productsold_release</i>) and install Mobile Administrator 9.8 (see <i>Installing webMethods and Intelligent Business Operations Products 9.8</i>)
Presto releases prior to 3.6	Contact Software AGGlobal Consulting Services
Terracotta Server Array when not used with webMethods products, or when setup is not a single server or a mirror group consisting of an active server and a mirror server	See Terracotta BigMemory Max documentation
Universal Messaging when not used with webMethods products	See Universal Messaging documentation

Release Numbering Exceptions

The products below did not follow general webMethods and IBO release numbering.

Product	Release Numbering
OneData	In webMethods 8.2, the OneData release number was 8.4.
Presto	In IBO 9.6, 9.7, and 9.8, the Presto release numbers were 3.7, 3.8, and 3.9, respectively.
Terracotta	In IBO 9.0, 9.5, 9.6, 9.7, and 9.8, the Terracotta release numbers were 3.7.4, 3.7.6, 3.7.6, 4.1.4, and 4.3, respectively.

Guides Needed to Perform Upgrades

You will always need the guides listed below to perform the upgrades in this guide.

- *Using the Software AG Installer (April 2015 release)*
- *Installing webMethods and Intelligent Business Operations Products 9.8*
- *Using the Software AG Update Manager*

Depending on the products you are upgrading, you might need the 9.8 guides and other guides listed below to perform the upgrades in this guide.

- *webMethods BPM Task Development Help*
- *Using Blaze Rules with BPM and CAF*
- *CentraSite documentation*
- *webMethods Deployer User's Guide*
- *webMethods Integration Server Administrator's Guide, webMethods Service Development Help, and webMethods Integration Server Clustering Guide*
- *Administering webMethods Optimize and Configuring BAM*
- *Presto documentation*
- *Getting Started with the webMethods Product Suite and Terracotta and Terracotta BigMemory Max documentation*

2 Critical Requirements For Successful Upgrade

- Critical Upgrade Requirements 16
- Product Requirements 16

Critical Upgrade Requirements

Important: If you do not conform to these requirements, you will experience unpredictable results, possibly including corruption of your installation and data. Software AG strongly recommends that you upgrade in a controlled test environment and test that environment for proper operation before upgrading your production environment.

- This chapter explains how to install 9.8 products in a fresh installation directory, prepare your old environment for upgrade, shut down your old products, and back up and then migrate your old database components to 9.8. You must perform these tasks in the exact order in which they are presented in this chapter.
- For some products, you must then perform additional tasks to complete the product upgrade. These tasks are described in subsequent chapters. You must perform the tasks in these chapters in the exact order in which the chapters and tasks are presented in this guide.
- For all products, you must perform all tasks in the final chapter in this guide after you have completed all other tasks.
- You must perform the tasks in this guide for all upgrade paths unless the guide notes that a particular task is required for certain upgrade paths only.
- You must upgrade a product from its latest service pack. For example, if the latest release of a product in release 8.2 was 8.2 SP3, you must upgrade that product from 8.2 SP3. You cannot upgrade the product from 8.2 SP2 or 8.2 SP1.
- In webMethods 9.7 and 9.8, the webMethods Broker release number is still 9.6 and is the same release that was available with webMethods 9.6. Even though the release is the same, you must still perform the tasks described in this chapter and this guide for webMethods Broker.

Product Requirements

API-Portal Requirements

You must have these functional privileges to upgrade API-Portal:

- License, user, and document management
- User management configuration
- Database administration

Ask your API-Portal administrator to log on to the User Management Component at `http://host:port/umc`, go to the user management page, click your user name, and either

add you to the API Administrator group or go to the Functional Privileges tab and assign the privileges listed above to you.

9.6 or 9.7 Upgrade: webMethods Broker Requirements

In webMethods 9.7 and 9.8, the webMethods Broker release number is still 9.6 and is the same release that was available with webMethods 9.6. If the existing webMethods Broker 9.6 from the webMethods 9.6 or 9.7 release is installed on its own machine, you can continue using it with your 9.8 products. However, you must still follow all instructions and perform all tasks for webMethods Broker in this chapter and in ["Complete the webMethods Broker Upgrade" on page 43](#).

If you want to upgrade to a machine that is running a different operating system or has different hardware than the old machine, or if you want webMethods Broker to exist with other 9.8 products in the same installation directory, you must install webMethods Broker 9.6 from the webMethods 9.8 release. You can install on a machine that is running the same or higher operating system and has the same hardware (for example, both have AMD, or both have Intel) as the old machine. Make sure you follow all instructions and perform all tasks for webMethods Broker in this chapter and in ["Complete the webMethods Broker Upgrade" on page 43](#).

Broker Server 9.6 is compatible with webMethods Broker 9.5, 9.0, and 8.2 command-line utilities; Broker client APIs for C, C#, Java, and JMS; and clients for JMS. Broker Server 9.5, 9.0, and 8.2 are compatible with webMethods Broker 9.6 command-line utilities, Broker client APIs for C, C#, and Java; and the webMethods Broker user interface in My webMethods.

CentraSite Requirements

If you want to install the CentraSite Application Server Tier and CentraSite Registry Repository on different machines, or in the same directory but at different times, you must install the Registry Repository first.

If you install CentraSite 9.8 on the same machine as the old CentraSite, you will not be able later to run both products at the same time, even if they use different ports, because of conflicts for IPC and shared memory resources.

Integration Server Clustering Requirements

If you are going to upgrade a cluster of Integration Servers, Software AG recommends the procedure below.

1. Install a matching set of 9.8 Integration Servers. For example, if you have a cluster of three old Integration Servers named IS1_old, IS2_old, and IS3_old, with an external load balancer to distribute client requests, install IS1_new, IS2_new, and IS3_new.
2. Perform the steps in ["Completing the Integration Server and Hosted Products Upgrade" on page 53](#) for each Integration Server in the cluster.

3. Shut down IS1_old.
 4. Migrate IS1_old data to IS1_new and customize the IS1_new startup files using the instructions in ["Completing the Integration Server and Hosted Products Upgrade" on page 53](#).
 5. Shut down IS2_old and IS3_old.
 6. Do the following for IS1_new:
 - a. Start IS1_new.
 - b. If IS1_old was not configured to use a Terracotta-based clustering solution, configure IS1_new to use a Terracotta-based clustering solution. Make sure IS1_new is processing client requests. For instructions, see *webMethods Integration Server Clustering Guide 9.8*.
- Important:** Support for Coherence was discontinued in release 9.0.
- c. Update WSDLs, connect functions to database components, and set up central user management on IS_new as described in ["Completing the Integration Server and Hosted Products Upgrade" on page 53](#).
 7. Repeat steps 4 and 6 for IS2_new and then for IS3_new.

Mobile Administrator Requirements

Mobile Administrator 9.8 requires a different license from previous releases. Obtain a Mobile Administrator 9.8 license file from Software AG.

My webMethods Server Requirements

For your old release, you might have identified a default My webMethods Server installation in the global path of the installation's host machine using the environment variable WM_HOME. If you are going to install the new release on the same machine as the old release, the existence of this variable could cause problems, so you must remove the environment variable from the global path. Do not redefine the environment variable after installing My webMethods Server 9.8.

The My webMethods Server migration utility allows you to migrate specific server instances within the old My webMethods Server installation to the new My webMethods Server installation. You only need to install one My webMethods Server 9.8 on each machine in the cluster.

3 Upgrading Products to the New Release

■ Read Critical Requirements for Successful Upgrade	20
■ Migrating Across Different Hardware or Operating Systems	20
■ Install the 9.8 Products	20
■ Prepare the Old Environment for Upgrade	22
■ Shut Down the Old Products	27
■ Migrate Database Components	27
■ Complete the Side-by-Side Upgrade	30

Read Critical Requirements for Successful Upgrade

Before performing any task in this guide, read "[Critical Requirements For Successful Upgrade](#)" on page 15 carefully and completely.

Migrating Across Different Hardware or Operating Systems

Software AG tests migration across machines that have the same hardware and operating systems. Integration Server also tests migration across machines that have different hardware or different operating systems. However, for other products, migration across machines that have different hardware or operating systems might have unpredictable results, especially when one machine is Windows and the other is non-Windows.

Install the 9.8 Products

If you are installing on the same machine as your old release, the installer often allows you to assign ports used by an old product to the new product as well, even if the old product is running. Assigning the same ports means you will not need to edit port values in assets and clients when you begin using the new release.

Follow the instructions in *Using the Software AG Installer* (April 2015 release) and *Installing webMethods and Intelligent Business Operations Products 9.8* to install your 9.8 products in a fresh installation directory, with the additional guidelines below.

1. For installation directory, specify a new Software AG installation directory.
2. On the product tree, select the 9.8 products to install. Also select Database Configuration and Update Manager.

Note: In webMethods 8.2, the Blaze design tool is Blaze Advisor. In 9.0, it is a Blaze plug-in to Software AG Designer. In 9.5 and later, for compatibility reasons, the Blaze plug-ins are delivered in a self-contained workbench that is based on Eclipse 3.6 and is separate from Software AG Designer 9.5 and later, which are based on Eclipse 4.x.

3. For the product panels, follow the instructions in *Installing webMethods and Intelligent Business Operations Products 9.8*, with the additional guidelines below.

Product	Guidelines
ActiveTransfer Server	Point the database connection at the ActiveTransfer database component you want to use with ActiveTransfer Server 9.8.

Product	Guidelines
webMethods Broker	Choose to <i>not</i> create a Broker Server configuration.
Integration Server	Point the database connection at the ISInternal database component you want to use with Integration Server 9.8.
My webMethods Server	<p>Choose to <i>not</i> create a My webMethods Server instance.</p> <p>Note: You will migrate old server instances later in this procedure. You will also migrate the My webMethodsServer database component you want to use with My webMethods Server 9.8, then point to that database component later in this procedure.</p>
OneData	<p>Enter the values you noted earlier for port, repository ID, and repository name, and database component connection. Make sure the database connections point to the OneData database components you want to use with OneData 9.8.</p> <p>Note: Reusing port values means that Web service clients of the old OneData will not have to change URLs to access OneData 9.8.</p>
Trading Networks	Point the database connection at the TradingNetworks database component you want to use with Trading Networks 9.8.
Universal Messaging	<p>Enter the name that matches the name of a server instance in the old Universal Messaging installation and bind to that server instance's IP address and port. Point the data directory to a location in the new installation (that is, do not point to the old data directory).</p> <p>Note: If you accept the default location for the data directory, it will be migrated automatically later. If you use a different location, you will have to migrate it manually.</p>

- After installation is complete, if you installed on a Windows system, and you installed products as Windows services, the default startup for the services is Automatic. Set the services to Manual. If you installed on a UNIX system, and you have scripts that automatically start daemons, disable the scripts.
- Follow the instructions in *Installing webMethods and Intelligent Business Operations Products 9.8* to complete the installation of your 9.8 products. This includes installing the latest product fixes. For instructions on installing fixes, see *Using the Software AG Update Manager* and the fix readme files.

Note: 9.6 or 9.7 upgrade: *Installing webMethods and Intelligent Business Operations Products 9.8* explains that you can create additional Integration Server instances after installation. However, you can also, or instead, have the Integration Server migration utility create 9.8 instances for you from the old instances (see "[Completing the Integration Server and Hosted Products Upgrade](#)" on page 53).

Important: Do not start any 9.8 products at this point, or before migrating database components or data. Do not start 9.8 products before the instructions in this guide tell you to do so, or your database components could become corrupted.

Prepare the Old Environment for Upgrade

Prepare Databases

Some products, like CentraSite, Integration Server, Monitor, Optimize, and Trading Networks, offer features to archive or purge data from product database components. You can reduce the amount of time needed to back up and migrate the product database components by archiving or purging them first.

Install Fixes on Products

Install the latest fixes on the old API-Portal, CentraSite, Infrastructure Data Collector, My webMethods Server, Optimize, and Process Engine. For instructions on installing fixes, see *Using the Software AG Update Manager* and the fix readme files. After you install a fix on a product, start the product.

Prepare the Old API-Portal

1. Start the old API-Portal Cloud Controller.
2. If the old API-Portal is installed in a clustered, high-availability setup, make sure all nodes are running and accessible to the ZooKeeper ensemble.
3. Register each node with the parent node by running this command:

```
acc> add node logical_node_name IP_address_or_host_name [:@port ]
user_name password
```

4. Run this command:

```
acc> startall
```

5. Back up the old master tenant to a file by running this command:

```
acc> set acc config backup.restore.tenant.app.types =
com.aris.umcadmin.y-umcadmin-run-prod
acc> set acc config backup.restore.tenant.app.type.names = umc
```

```
acc> backup tenant master to full_path_to_backup_file
username=your_user_name password=your_password
```

- For each tenant in the old API-Portal, you will have to create a tenant in API-Portal 9.8. List all tenants in the old API-Portal by running this command:

```
acc> list tenants
```

- Back up each old tenant's API and related data; user data; API-Portal document storage data, including all access rights; and collaboration data to a file by running the commands below. The commands will add the file extension .acb to each file.

Important: Inform API-Portal users that they cannot work on the tenants during this backup process.

```
acc> set acc config backup.restore.tenant.app.types =
com.aris.umcadmin.y-umcadmin-run-prod, com.aris.modeling.components.
y-server-run-prod, com.aris.adsadmin.y-adsadmin-run-prod,
com.aris.runnables.ecp-run-prod
acc> set acc config backup.restore.tenant.app.type.names = umc, abs, ads, ecp
acc> backup tenant tenant_name to full_path_to_backup_file
username=your_user_name password=your_password
```

- If you installed API-Portal 9.8 on the same machine as the old API-Portal, run this command:

```
acc> stopall
```

Prepare the Old CentraSite

- If you changed settings for CentraSite Control from their defaults, note those settings so you can make the same changes in the new CentraSite installation.
- 8.2 upgrade: The metamodel that defines the internal structure of ApplinX object types changed in release 9.0 and is not compatible with the old metamodel. CentraSite 9.8 does not support the use of object type instances that are based on the old ApplinX object types. CentraSite 9.8 does not allow ApplinX object types or instances from old product releases to be imported, and there is no procedure to migrate old instances to the new metamodel. If you want to use ApplinX object type instances from the old product release in CentraSite 9.8, you will have to manually re-create the instances after you install CentraSite 9.8. Note the definitions of all your ApplinX object type instances.

Prepare the Old Software AG Designer

Export Integration Server Definitions

- In the old Software AG Designer, go to **Window > Preferences**. On the **Preferences** dialog box, in the left navigation bar, go to **Software AG > Integration Servers**.
- Click **Export** and complete the dialog box. Software AG Designer will save the file with the extension .properties.

Export CloudStreams Server Definitions and Projects

1. To export CloudStreams Server definitions, go to **Window > Preferences**. On the **Preferences** dialog box, in the left navigation bar, go to **Software AG > CloudStreams Servers**. Click **Export** and complete the dialog box. Software AG Designer will save the file with the extension `.properties`.
2. Open the CloudStreams Development perspective. In the CloudStreams Governance list, right-click **CloudStreams Governance projects to export**, click **Export**, and complete the dialog box.
3. Repeat the previous step to export CloudStreams Provider projects.

Export Event Bus Console Configurations

1. In the old Software AG Designer, open the **File > Export** wizard. In the Select panel, go to **Run/Debug > Launch Configurations** and click **Next**.
2. In the tree, select the Event Bus Console configurations. In the **Location** field, specify the directory to which to export. Software AG Designer will save each file with the extension `.launch`.

Export Preferences

If you want to migrate your preferences to Software AG Designer 9.8, you export them from the old Software AG Designer and then import them into Software AG Designer 9.8. To export the preferences, do the following:

1. In the old Software AG Designer, open the **File > Export** wizard. In the Select panel, go to **General > Preferences** and click **Next**.
2. In the **To preference file** field, specify the name of the file to which you want to export your preferences. Software AG Designer will save the file with the extension `.epf`.

Prepare the Old Integration Server

1. If you are using webMethods Broker, shut down the new Broker Monitor if it is running. Restart the old Broker Monitor, and then start the old Broker Server.
2. Start the old Integration Server.
3. Suspend document retrieval for all triggers and make sure all guaranteed messages have finished processing, as follows:
 - a. Open Integration Server Administrator and point to the old Integration Server.
 - b. Go to **Packages > Management**. Click **View Locked Elements**, click **Unlock Elements**, select all elements of triggers, and then click **Unlock Selected Elements**.
 - c. 8.2 or 9.0 upgrade: Go to the **Settings > Messaging > Broker/Local Trigger Management** page. In the **Individual Trigger Controls** area, in the **Active** column under **Document**

- Retrieval**, click **edit all** and set **Retrieval State** to **Suspended**. Refresh the page until the **Persisted Queue Counts** field shows 0 for every trigger.
- d. 9.5, 9.6, or 9.7 upgrade: Go to the **Settings > Messaging > webMethods Trigger Management** page. In the **Individual Trigger Controls** area, in the **Active** column under **Document Retrieval**, click **edit all** and set **Retrieval State** to **Suspended**. Refresh the page until the **Current Queue Counts** field shows 0 for every trigger.
4. If you are using webMethods Broker, make sure all outbound Broker messages in the client side queue have been sent, as follows:
 - a. Make sure Integration Server is connected to the Broker.
 - b. 8.2, 9.0, or 9.5 upgrade: Go to the **Settings > Resources > Store Settings** page. In the **Outbound Document Store** area, make sure the **Current Documents in Outbound Store** field shows 0.
 - c. 9.6 or 9.7 upgrade: Go to the **Settings > Messaging > webMethods Messaging Settings** page. Make sure the **CSQ Count** field shows 0 for the Broker connection alias.
- Note:** The **CSQ Count** field contains the same information as the **Current Documents in Outbound Store** field on the **Settings > Resources > Store Settings** page. You need only check the **CSQ Count** field.
5. 9.6 or 9.7 upgrade: If you are using Universal Messaging, make sure all outbound Universal Messaging messages in the client side queue have been sent, as follows:
 - a. Make sure Integration Server is connected to each Universal Messaging server that is acting as a webMethods messaging provider.
 - b. Go to the **Settings > Messaging > webMethods Messaging Settings** page. Make sure the **CSQ Count** field shows 0 for the Universal Messaging connection aliases.
 6. Make sure all outbound JMS messages in the client side queue have been sent, as follows:
 - a. Make sure Integration Server is connected to the JMS providers.
 - b. Go to the **Settings > Messaging > JMS Settings** page. In the **JMS Connection Alias Definitions** area, make sure the **CSQ Count** field shows 0 for every JMS connection alias.
 7. If you are going to upgrade business processes, and you have business process models that use volatile transition documents and process instances of those models are in a Started state, allow the Started processes instances to complete or suspend them before you upgrade.

Important: If you do not allow Started process instances to complete or suspend before you upgrade, you could lose data.

Note: Started process instances of models that use guaranteed transition documents do not have to complete or be suspended before you upgrade.

Prepare the Old OneData

1. In the old OneData, go to the **Home > Administer > Job Center** page. On the **Filter** page, for each job type, filter by **Active** or **Pending Active**. If a job has one of these statuses, terminate the job or wait for it to complete.
2. In the file system, go to the `old_Software AG_directory \profiles\ODE\configuration \tomcat\conf` directory and open the `context.xml` file. Note the connection prefix located at the beginning of the schema user name (for example, in the metadata schema user name `dev2_md`, the connection prefix is `dev2`). Also note the database connection parameters for the metadata, work area, and release area database schemas.
3. Note the port numbers, as follows:
 - a. 8.4 upgrade: Go to the `old_Software AG_directory \profiles\ODE\configuration \tomcat\conf` directory and open the `server.xml` file. Note the HTTP and HTTPS port numbers in the port attribute of the `Default CTP HTTP Connector` and `Default CTP HTTPS Connector` files, respectively.
 - b. 9.0, 9.5, 9.6, or 9.7 upgrade: Go to the `old_Software AG_directory \profiles \ODE\configuration\com.softwareag.platform.config.propsloader` director. Open the `com.softwareag.catalina.connector.http.pid-port.properties` and `com.softwareag.catalina.connector.https.pid-port.properties` files and note the port numbers.

Prepare the Old Optimize

1. If you want to reuse your Central Configuration System (CCS) Environment definition, export the environment to an XML file. In My webMethods, go to the **Administration > System-Wide > Environments > Define Environments** page. Select the check box next to the environment, click **Export**, and provide a file name and location for the exported environment.
2. 8.2 upgrade: If you have Integration Servers that are clustered using Oracle Coherence, you will reconfigure them later in this guide to use a Terracotta-based clustering solution instead. Prepare for this change as follows:
 - a. If you are monitoring the clustered Integration Servers, stop monitoring them.
 - b. If you have KPIs for the clustered Integration Servers, go to the **Applications > Administration > Analytics > Infrastructure Components > Monitored Components** page. In the Monitored Components list, click **IS Cluster**. Note the KPIs listed in the **Selected KPIs** box, delete the KPIs, and then click **Save**.

Note: KPI data that was gathered from the clustered Integration Servers will not be available in Optimize 9.8.

3. If you want to cluster Analytic Engines in the new installation, you will need a Terracotta Server Array. For planning information, see *Getting Started with the*

webMethods Product Suite and Terracotta 9.8, Configuring BAM 9.8, and the Terracotta BigMemory Max 4.1 documentation.

Shut Down the Old Products

Shut down your old products. For a UNIX system, use the instructions in the product documentation for your old release. For a Windows system:

Products running as...	Shut down from...
Services	Windows Services window. Services are listed as Software AG <i>product release</i> .
Applications	Windows Start menu. Applications are listed as Software AG > Stop Servers > product .

If you are upgrading CentraSite, and the old and new CentraSite are on the same machine:

- 8.2 upgrade: Shut down all old Application Server Tiers by stopping the CentraSite Apache and Tomcat Server services.
- 9.0, 9.5, 9.6, or 9.7 upgrade: Shut down all old Application Server Tiers by stopping the Software AG Runtime services.
- Shut down the old Registry Repository by stopping its service.

If you are upgrading EntireX, also shut down all Brokers, RPC Servers, and customer applications that use EntireX libraries. For instructions, see the product documentation for your old release.

If you are upgrading Mobile Administrator, stop it by running the command `/etc/init.d/appfresh-mobile stop`

If you are upgrading a cluster of My webMethods Servers, shut down all cluster nodes. A cluster is defined as multiple nodes that point to the same MywebMethodsServer database component. Nodes can be My webMethods Server installations on different machines or server instances within a single My webMethods Server installation.

Migrate Database Components

Back Up the RDBMSs

Back up the RDBMSs that contain your database components using the vendor instructions for your RDBMS.

If you are upgrading My webMethods Server, back up the My webMethods Server installation directory at the same time you back up the MywebMethodsServer database component. If you have problems, you will need data from the backup to restore the database.

Upgrade the RDBMSs

Check your RDBMSs against *webMethods and Intelligent Business Operations System Requirements* 9.8. If the RDBMS version you are using is not supported by your 9.8 products, you must upgrade to a supported RDBMS version before you continue with the instructions in this chapter.

Shut Down Products that Connect to Database Components

If you have not already done so, shut down all old Integration Server, My webMethods Server, OneData, Optimize, and Presto instances that connect to database components.

Before You Migrate Trading Networks Database Components

1. Use your data editor to delete all data from the ARCHIVE_WORKTABLE table in the TradingNetworksArchive database component.
2. If you created custom indexes for your Trading Networks database components, check whether those custom indexes conflict with indexes that will be created when you run the Trading Networks migration scripts in the next step. If so, drop the custom indexes. The migration scripts are located in the *9.8_Software AG_directory \common\db\scripts\RDBMS \tradingnetworks \migrate* directory, under the directories listed below.

Old Release	Directories
8.2	\39_40, 40_to_45, and 45_to_50
9.0	\40_to_45 and 45_to_50
9.5 or 9.6	\45_to_50
9.7	\50_to_55

Special Considerations for Migrating Other Database Components

- If you are using business processes, you have stored information about in-progress, completed, or failed business process instances. This stored information is called business process run-time data. If you do not want to finish running in-progress

process instances or resubmit completed or failed process instances from your old installation in webMethods and IBO 9.8, do not migrate your business process run-time data (see table below).

- If you are using Optimize with a DB2 RDBMS, there are special considerations for the URL you specify on the database component migration command. You must do the following:
 - Specify the schema name in the URL using all capital letters.
 - Specify the options `CreateDefaultPackage=true`, `ReplacePackage=true`, and `DynamicSections=3000`. These settings will affect all database components in the same schema or database.

Install Fixes on Database Scripts

Install the latest fix on the old database scripts using the Software AG Update Manager. The names for database script fixes follow the convention `DC_old_release_number_DBS_Fixlatest_fix_number`. For instructions on installing fixes, see *Using the Software AG Update Manager* and the fix readme file.

Database Component Names

Database component names you might use in the migration command are as follows:

If you are migrating this data...	Specify these <i>database_component</i> values
ActiveTransfer Server	ActiveTransfer
Business process run-time	ProcessAudit, ProcessEngine, Archive, Staging, Reporting
Business rules	BusinessRules
Integration Server	ISInternal, ISCoreAudit, CrossReference, DocumentHistory, DistributedLocking
My webMethods Server	MywebMethodsServer
OneData	OneDataMetadata, OneDataReleaseArea, OneDataWorkArea
Optimize	Analysis, ProcessTracker, ProcessAudit, DataPurge

If you are migrating this data...	Specify these <i>database_component</i> values
Trading Networks	TradingNetworks, TradingNetworksArchive

Migrate Database Components

Important: After you migrate database components to 9.8, you can no longer use them with your old environment.

On the machine on which you installed Database Configuration 9.8, open a command window or shell, go to the *9.8_Software AG_directory \common\db\bin* directory, and migrate database components by running the commands below. After you run each command, check the log file *dcc_yyyymmddhhmmss* in the *9.8_Software AG_directory \common\db\logs* directory.

- If you want to migrate all your database components, run this command:

```
dbConfigurator.{bat|sh} -a migrate -d {oracle|sqlserver|db2luw}
-c all -v latest -l db_server_URL -u existing_db_user -p password
```

If you are using an Oracle or DB2 RDBMS, and you are not using the default tablespace, also specify the *-tsdata data_tspace_name* and *-tsindex index_tspace_name* parameters.

Note: Since you most likely do not have all database components (because you do not have all products), you will see messages in the log file that say migration failed for the database components you do not have. You can ignore these messages.

- If you do not want to migrate all your database components, run this command for each database component you need to migrate:

```
dbConfigurator.{bat|sh} -a migrate -d {oracle|sqlserver|db2luw}
-c database_component_name -v latest -l db_server_URL
-u existing_db_user -p password
```

Complete the Side-by-Side Upgrade

To complete the side-by-side upgrade procedure, perform the steps in the product-specific chapters that follow. For many products, the steps you perform in these chapters include running a migration utility that automatically migrates data from the old installation to the new installation. The section below describes the general behavior of the migration utilities. Any exceptions to the general behavior are noted in the product-specific chapters.

Migration Utilities

Some migration utilities for some products run without prompting you for any information. Migration utilities for other products ask you to select the data to migrate, and offer these options:

- You can run a custom migration, in which you select the data to migrate. The utility gathers your settings through a series of prompts, then migrates the selected data. You can export your settings and use them in other upgrades.
- You can run a migration with imported settings. The imported settings can come from settings you exported from a custom migration, or from the default migration provided with the product installation. The settings for default migrations are described in the product-specific chapters.

Note: When you run migration utilities, you provide the full path to the old installation, and sometimes the path to the 9.8 installation. If you supplied a symbolic link as the installation directory when you installed the old or 9.8 product, the path you provide to the migration utility must be the same symbolic link path you supplied during installation.

If a migration utility fails to migrate a piece of data, the utility asks whether to continue with the next piece of data or abort the migration. If you choose to abort, the utility exits. The utility does not revert the 9.8 product installation. You can address the issue and rerun the utility.

Migration utilities write detailed migration information to the command window and to the migrationLog.txt file in the *9.8_Software AG_directory \install\logs* directory. By default, utilities write INFO, ERROR, and FATAL messages to the log. If you want to increase the logging level for a product's migration to DEBUG, go to the product directory that contains the log4j.properties file (for example, the *9.8_Software AG_directory \product \bin \migrate* or *\migrate \bin* directory), open the file in a text editor, set the `log4j.logger.Migration` property to `DEBUG, MigrationFile`, and save and close the file.

Use of Zip File as Source of Data to Migrate

If your old and new installations for a product are on different machines, migration utilities can use a Zip file of the old product installation as the source of the data to migrate.

Create a Zip File for My webMethods Server

1. Go to the *9.8_Software AG_directory /MWS/bin/migrate* directory and copy the `zip-mws.{bat|sh}` file to the *8.2_Software AG_directory /MWS/bin* directory or the *9.0_or_later_Software AG_directory /MWS/bin/migrate* directory.

Note: If you installed fixes on the old My webMethods Server, the file might already exist.

2. On the old machine, open a command window or shell, go to the directory that contains the file you just copied, and run the command `zip-mws.{bat|sh}`. The command creates a zip file named `mws.zip` in `old_Software AG_directory /MWS/migrate-zip` directory.
3. Copy the `mws.zip` file to any directory on the 9.8 machine that hosts My webMethods Server.

Important: If using FTP to copy, use the binary file transfer mode/type. If you use another mode/type, the Zip file might become corrupted.

Create a Zip File for All Other Products

The instructions below use the Java Archive tool to create the Zip file. Specify the location of the Java Archive tool in the `JAVA_HOME` and `PATH` system variables on the machine that hosts the old product installation. The tool is located in the `Software AG_directory \jvm\jvm\bin` directory.

Note: On some systems, the lower-level `jvm` directory name includes additional information, such as `\jvm\jvm160_32`, or `\jvm\jvm170`, or `\jvm\jvm_64`.

To create the Zip file for a product, go to the product's old machine and open a command window or shell. Go to the directory indicated in the table below and enter the command indicated in the table. If multiple commands are listed, use the same Zip file name in each command.

Product	Go to this directory...	Run this command...
Command Central	Software AG directory that contains old Command Central	<pre>jar cfM Zip_file_name.zip common \conf jar ufM Zip_file_name.zip profiles\CCE jar ufM Zip_file_name.zip install \products</pre>
EDA/NERV	Software AG directory that contains old EDA/NERV	<pre>jar cfM Zip_file_name.zip install \products jar ufM Zip_file_name.zip common\conf jar ufM Zip_file_name.zip common \EventTypeStore jar ufM Zip_file_name.zip common\nerv \bundles</pre>

Product	Go to this directory...	Run this command...
		<p>The common\nerv\bundles directory will exist only if you deployed NERV bundles in the old release.</p> <p>If you deployed NERV bundles or the Event Type Store to non-default locations, add the bundles to the Zip file to preserve the file structure shown in the commands above.</p>
Infrastructure	Software AG directory that contains old Software AG Runtime	<pre>jar cfM Zip_file_name.zip common \conf install\products profiles \CTP</pre> <p>You might see the message profiles\CTP: no such file or directory. You can ignore this message.</p>
Infrastructure Data Collector	Old Infrastructure Data Collector directory	<pre>jar cfM Zip_file_name.zip common \conf install\products profiles \InfraDC</pre>
Integration Server	Old Integration Server directory	<pre>jar cfM Zip_file_name.zip *</pre>
OneData	Software AG directory that contains old OneData	<pre>jar cfM Zip_file_name.zip profiles/ODE/bin/onedata/config/*</pre>
9.6 or 9.7 upgrade: Platform Manager	Software AG directory that contains old Platform Manager	<pre>jar cfM Zip_file_name.zip common \conf jar ufM Zip_file_name.zip profiles\SPM jar ufM Zip_file_name.zip install \products</pre>
Universal Messaging	Old Universal Messaging directory	<pre>jar cfM Zip_file_name.zip server lib</pre>

Copy the product's Zip file to any directory on the 9.8 machine that hosts the product.

Important: If using FTP to copy, use the binary file transfer mode/type. If you use another mode/type, the Zip file might become corrupted.

4 Complete the webMethods and IBO Infrastructure Upgrade

Upgrade

■ Overview	36
■ Before Migrating Data	36
■ Run the Infrastructure Migration Utility	36
■ 9.0, 9.5, 9.6, or 9.7 Upgrade: Update the Software AG Runtime NERV Configuration	37
■ Complete Final Upgrade Tasks	37

Overview

Infrastructure components are automatically installed with many but not all products. Determine whether the files referenced in this chapter exist on your machine. If they do, complete the tasks in this chapter.

Important: The migration utility supports migration from a Windows system to another Windows system, and from a UNIX system to another UNIX system that is using a JVM from the same vendor. The utility does not support migration from one operating system to a different operating system (for example, does not support Windows system to a UNIX system, or vice versa). For a list of operating systems supported by the 9.8 release of webMethods and Intelligent Business Operations, see *webMethods and Intelligent Business Operations System Requirements*.

Before Migrating Data

1. Make sure all products in the old and new installations are shut down.
2. Install the latest fixes on the 9.8 Infrastructure. The fix names will look like wMFix.OSGI.Migration. For instructions, see *Using the Software AG Update Manager* and the fix readme files.

Run the Infrastructure Migration Utility

The Infrastructure utility migrates the following:

- The infrastructure configuration.
- If present, the Software AG Runtime configuration.
- 8.2 upgrade: JAAS and port configurations, and the user repository.
- 9.0, 9.5, 9.6, or 9.7 upgrade: Debug, JAAS, JMX, port, security, single-sign on, watchdog, and Web Services Stack configurations; journal log settings; the user repository; and the password store.
- 9.5, 9.6, or 9.7 upgrade: Users, groups, and roles.

The Infrastructure migration utility runs without prompting you for any information. If an error occurs, the utility exits.

For a general description of migration utility behavior, see "[Migration Utilities](#)" on page 31.

1. On the 9.8 machine, open a command window or shell, go to the `9.8_Software AG_directory \common \migrate \osgi \bin` directory, and run this command:

```
migrate.{bat|sh} {-srcDir|-srcFile} full_path_to_{old_Software  
AG_directory|Zip_file} -destDirfull_path_to_9.8_Software  
AG_directory -silent true
```

2. If you see a *Software AG_directory \profiles\CTP* directory in both the old and the 9.8 installation directories, go to the *9.8_Software AG_directory \profiles\CTP\bin \migrate* directory and run the same command again.

9.0, 9.5, 9.6, or 9.7 Upgrade: Update the Software AG Runtime NERV Configuration

Go to the *9.8_Software AG_directory \profiles\CTP\configuration \com.softwareag.platform.config.propsloader* directory and open the *com.softwareag.platform.eda.nerv.properties* file. Make sure the JMS provider references point to your 9.8 JMS provider and the file system paths point to the 9.8 installation. If they do not, update them.

9.5, 9.6, or 9.7 upgrade: Open the same file in the old installation. If any values in the two files other than the path values mentioned above are different, copy the old value over the 9.8 value.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in ["Complete Final Upgrade Tasks for All Products" on page 118](#).

5 Complete the EDA/NERV, Terracotta, and Messaging Upgrades

■ Complete the EDA/NERV Upgrade	40
■ Complete the Terracotta Upgrade	40
■ Complete the Universal Messaging Upgrade	41
■ Complete the webMethods Broker Upgrade	43

Complete the EDA/NERV Upgrade

Run the EDA/NERV Migration Utility

The EDA/NERV migration utility migrates the following:

- 9.0, 9.5, 9.6, or 9.7 upgrade: Software AG Designer Event Bus Console connection configurations that use NERV.
- 9.0, 9.5, 9.6 or 9.7 upgrade: NERV configuration bundles, if you deployed any.
- Event Type Store.

The EDA/NERV migration utility runs without prompting you for any information. If an error occurs, the utility exits.

For a general description of migration utility behavior, see ["Migration Utilities" on page 31](#).

On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory* \common\migrate\EDA\bin directory and run this command:

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory|Zip_file}
-destDir full_path_to_9.8_Software AG_directory
-importFile 9.8_Software AG_directory\common\migrate\EDA
\bin\migrate{822|901|951|960|970}sbs.dat
-silent true
```

Update Paths

Go to the *9.8_Software AG_directory* \common\conf directory and open the com.softwareag.eda.nerv.properties file. Verify that the JMS provider references point to your 9.8 JMS provider and the update file system paths to point to the 9.8 installation. If they do not, update them.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in ["Complete Final Upgrade Tasks for All Products" on page 118](#).

Complete the Terracotta Upgrade

This section explains how to complete an upgrade of a Terracotta Server Array that has a single server or a mirror group consisting of an active server and a mirror server. For all other setups, see the Terracotta BigMemory Max 4.3 documentation.

Follow the instructions in *Getting Started with the webMethods Product Suite and Terracotta* to install the 4.1.4 or later license key and tc-config.xml file.

If you have a single server, shut down your old Terracotta Server Array and start the 4.3 Terracotta Server Array.

If you have a mirror group consisting of an active server and a mirror server, shut down the old mirror server and then shut down the old active server. Start active server 4.3 and then start mirror server 4.3.

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

Complete the Universal Messaging Upgrade

Before Migrating Data

Make sure all old Universal Messaging server instances are shut down. If any server instances are running in the 9.8 installation (for example, the default server instance umserver), make sure those are shut down as well.

Run the Universal Messaging Migration Utility

The Universal Messaging migration utility creates a 9.8 server instance on each server instance in the old installation and migrates the old instance's data directory and license file to the 9.8 server instance.

For a general description of migration utility behavior, see "[Migration Utilities](#)" on page 31. The Universal Messaging migration utility does not offer a default migration.

Perform a Custom Migration

1. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \UniversalMessaging\tools\migrate* directory, and run the command `migrate.{bat|sh}`.
2. The utility asks for the full path to one of the following:
 - The old Universal Messaging installation (for example, *C:\old_Software AG_directory \nirvana* or *UniversalMessaging* or */opt/old_Software AG_directory /nirvana* or *UniversalMessaging*).
 - The Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter N.
4. The utility asks whether to migrate each old server instance. If the utility cannot determine whether the instance was running as an application or service, it asks for that information as well.

If the data directory or license file for an old server instance is not in the default location (*Software AG_directory \nirvana* or *UniversalMessaging \server \server_instance \data* and *Software AG_directory \nirvana* or *UniversalMessaging \server \server_instance*, respectively), or the utility cannot determine where the data directory or license file is located, you will have to migrate them manually after the migration utility finishes, as explained later in this procedure.

If a server instance already exists in the 9.8 installation, the utility does the following:

- If the 9.8 instance has the same name as an old instance, the utility asks whether to migrate the old instance over the 9.8 instance. If you enter Y, the utility backs up the 9.8 instance to the *9.8_Software AG_directory \UniversalMessaging \migrationbackup* directory before migrating the old instance.
 - If the 9.8 instance has a different name than any of the old instances, the utility ignores the 9.8 instance.
5. The utility asks whether to export your settings. If you want to perform other migrations by importing the settings from this session, enter Y. If not, enter N.

Migrate Using Imported Settings

Imported settings come from settings you exported from a custom migration. These settings are stored in a file named *migrate.dat* in the *9.8_Software AG_directory \UniversalMessaging \tools \migrate* directory from which you ran the custom migration. Copy the *migrate.dat* file to any directory on machines that host Universal Messaging 9.8 installations to which you want to migrate data.

Migrating Using Imported Settings with Prompting

1. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \UniversalMessaging \tools \migrate* directory, and run the command `migrate.{bat|sh}`.
2. The utility asks for the full path to the old Universal Messaging installation or to the Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter Y and, when prompted, provide the full path to the *migrate.dat* file.

Migrate Using Imported Settings without Prompting

On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \UniversalMessaging \tools \migrate* directory, and run the command below. If an error occurs, the utility exits.

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Universal_Messaging_directory|Zip_file}
-importFile full_path_to_migrate.dat -silent true
```

Migrate Custom Changes, Data Directory, and License File

1. In release 9.8, the `nserver.conf` and `nserverdaemon.conf` files were combined into a single file named `Server_Common.conf`. If you made any custom changes to the `nserver.conf` or `nserverdaemon.conf` file for an old server instance, make the same changes to the `Server_Common.conf` file for the corresponding 9.8 server instance.
2. If the data directory for an old server instance was not in the default location specified in the Software AG Installer, the migration utility did not migrate the data directory. Copy the data directory from the old server instance to the 9.8 server instance. The default location for the data directory is the `9.8_Software AG_directory \ UniversalMessaging \ server \ server_instance \ data` directory. Then edit the file to point to the new server instance's `Server_Common.conf` file in that directory.
3. If the license file for an old server instance was modified or moved from the default location it was placed in by the Software AG Installer, the migration utility did not migrate the license file. Copy the license file from the old server instance to the 9.8 server instance. The default location for the license file is the `9.8_Software AG_directory \ UniversalMessaging \ server \ server_instance \ data` directory. Then edit the file to point to the new server instance's `Server_Common.conf` file in that directory.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in ["Complete Final Upgrade Tasks for All Products" on page 118](#).

Complete the webMethods Broker Upgrade

Overview

8.2, 9.0, or 9.5 upgrade: Follow the instructions in ["Create Broker Server 9.6 and Use the Old Storage" on page 44](#).

9.6 or 9.7 upgrade:

- If you continued using your webMethods Broker 9.6 from the webMethods 9.6 or 9.7 release, point your Broker clients, including Integration Server 9.8, to the old webMethods Broker 9.6 installation.
- If you installed webMethods Broker 9.6 from the webMethods 9.8 release, follow the instructions in ["Create Broker Server 9.6 and Use the Old Storage" on page 44](#).

If you installed webMethods Broker 9.6 on a machine that is running a different operating system or has different hardware than the old machine, or on an IBM System z machine that is running SUSE Linux Enterprise Server or Red Hat Enterprise Linux,

create new storage and then migrate Broker Server configuration information from the old storage to the new storage using the webMethods Broker export/import utilities. For instructions, see *Administering webMethods Broker 9.6*.

If you want to migrate from webMethods Broker to Universal Messaging, see *Migrating from webMethods Broker to webMethods Universal Messaging 9.8*.

Create Broker Server 9.6 and Use the Old Storage

1. Back up the old data directory.
2. Go to the *old_Software AG_directory \Broker\bin* directory and back up the old awbrokermon.cfg file.
3. Open the awbrokermon.cfg file and note the location of the old data directory as specified on the dataDir parameter.
4. Stop the old Broker Server, go to the *old_Software AG_directory \Broker\bin* directory, and run this command:


```
server_config remove full_path_to_old_data_directory
```

When asked whether to remove the Broker Server from the Broker Monitor configuration, enter Y.
5. If you need to copy the data directory to a new location (for example, because the old and new webMethods Broker installations are on different machines), do the following:
 - a. Copy the old data directory to the new location. If any storage files were located outside the data directory (for example, .stor and .data files), also copy those files.
 - b. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \Broker\bin* directory, and run the command below. If you also copied old storage files, specify the qs_map_file parameter for each file.


```
server_config relocate full_path_to_new_data_directory  
[-qs_map_file full_path_to_old_file full_path_to_new_file]
```
6. Create a Broker Server 9.6. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \Broker\bin* directory, and run the command below. Broker Server 9.6 details are automatically added to the Broker Monitor 9.6 startup configuration. Make sure the host name and port number for Broker Server 9.6 are the same as they were for the old Broker Server.


```
server_config add full_path_to_old_data_directory -k 9.6_license_file  
-p old_release_port
```

Note: If you are migrating on multi-home machines, and the command fails due to a port binding error, see ["Migrate on Multi-Home Machines" on page 45](#).
7. Make sure Broker Server 9.6 is running as follows:

System	Steps
Windows	<ol style="list-style-type: none"> Open the Windows Services window and make sure the status of the Software AG Broker Server 9.6(<i>port</i>) service is Started. Right-click the service and click Properties. Make sure the Path to Executable field points to the awbroker file for 9.6.
UNIX	<ol style="list-style-type: none"> Run the command <code>ps -ef grep awbroker</code>, which prints all running Broker processes including the path to the executable. Make sure the output text shows a line like this: <pre>full_path_to_9.8_Software_AG_directory/Broker/bin/awbroker -d full_path_to_old_data_directory</pre> Make sure the awbroker process is running and that it is running from the 9.6 awbroker file.

Migrate on Multi-Home Machines

If you are migrating from and to multi-home machines (that is, machines that have multiple network interfaces and IP addresses), the `Brokerserver_config add` command will disregard the migrated Broker Server settings from the old release, ignore specific IP bindings, and try to bind to the IP address 0.0.0.0. If the command fails due to a port binding error, follow the steps below.

- Go to the `9.8_Software_AG_directory/Broker/bin` directory, open the `awbrokermon.cfg` file, and add the entries below.

```
server-6849.datadir=full_path_to_data_directory
server-6849.exec=Software_AG_directory/Broker/bin/awbroker
server-6849.version=9.6
server-6849.port=6849
```

- Go to the `9.8_Software_AG_directory/Broker/data/awbrokerrelease_number/default` directory, open the `awbroker.cfg` file, and add the entries below.

```
executable=9.8_Software_AG_directory/Broker/bin/awbroker
version=9.6
```

- Start the Broker Monitor and use the `broker_status` command to make sure the Broker Server is running and contains history statistics.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

6 Completing the My webMethods Server Upgrade

■ Before Migrating Data	48
■ Run the My webMethods Server Migration Utility	48
■ 8.2 Upgrade: Migrate the My webMethods Server Infrastructure	50
■ 9.0, 9.5, 9.6, or 9.7 Upgrade: Migrate the My webMethods Server Infrastructure	51
■ Initialize My webMethods Server	51
■ Update Host Names	51
■ Complete Final Upgrade Tasks	52

Before Migrating Data

1. Make sure the old My webMethods Server is shut down.
2. If you are upgrading a cluster of My webMethods Servers, make sure all cluster nodes are shut down before migrating data for any My webMethods Server in the cluster. A cluster is defined as multiple nodes that point to the same MywebMethodsServer database component. Nodes can be My webMethods Server installations on different machines or server instances within a single My webMethods Server installation.

Run the My webMethods Server Migration Utility

The My webMethods Server migration utility migrates server instances within the old installation to the new installation.

For a general description of migration utility behavior, see "[Migration Utilities](#)" on page 31. Exceptions:

- The My webMethods Server migration utility does not offer a default migration.
- The log4j.properties file for My webMethods Server is located in the *9.8_Software AG_directory \MWS\bin\migrate\resources* directory.

Perform a Custom Migration

1. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \MWS\bin\migrate* directory, and run the command `migrate.{bat|sh}`.
2. The utility asks for the full path to one of the following:
 - a. The Software AG directory that contains the old My webMethods Server installation (for example, *C:\old_Software AG_directory* or */opt/old_Software AG_directory*).
 - b. The Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter N.
4. The utility asks about migrating server instances. If the installation has multiple server instances, the utility will ask separately about each instance.
 - a. The utility asks whether to migrate one of the server instances in the old My webMethods Server installation. Enter Y.

Note: If the server instance name already exists in the My webMethods Server 9.8 installation directory, you can choose to delete the 9.8 server

instance so migration of the old server instance can proceed, or to skip migration of this instance.

- b. The utility asks for the MywebMethodsServer database component to use with the migrated server instance. If you choose to use a copy of the production database, the utility prompts for the database URL, user name, and password.

Note: You cannot use a new database that contains no data.

5. The utility asks whether to export your settings from this session. If you want to perform other migrations by importing the settings from this session, enter Y. If not, enter N.
6. The utility asks whether to begin migration. If you enter Y, the utility migrates the data you selected. If migration completes successfully, the utility also does the following:
 - Deletes old configuration files that are not used by My webMethods Server 9.8, and deletes old properties that are not used by My webMethods Server 9.8 from the 9.8 configuration files.
 - If the old My webMethods Server used the embedded database, copies the database tables from the old My webMethods Server to My webMethods Server 9.8 and converts the tables to the new format.
7. 8.2 upgrade: If you are using CAF, and you added custom Jetty settings to the 8.2 mws-web.xml file for one or more CAF applications, the migration utility migrated those files. However, the schema and the name of the WebApplicationContext class in those files changed starting in 9.0, so you must manually migrate your custom Jetty settings.

For each CAF application, open the 8.2 mws-web.xml file and the 9.8 jetty-mws.xml file in the 8.2 and 9.8 WEB-INF directories. Copy the 8.2 custom Jetty settings to the Jetty section of the 9.8 file. Then configure the copied settings using the new schema and the name of the WebApplicationContext class format. The following example shows how the Jetty setting maxFormContentSize was configured in 8.2, and how it should be configured in 9.8. The format for the rest of the custom Jetty settings will be similar.

Sample configuration setting in My webMethods Server 8.2:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Configure PUBLIC "-//Mort Bay Consulting//DTD Configure//EN"
"http://jetty.mortbay.org/configure.dtd">
<Configure class="com.webmethods.portal.webapp.jetty6.MwsWebApplicationContext">
<Set name="maxFormContentSize">500000</Set>
</Configure>
```

Sample configuration setting in My webMethods Server 9.8:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Configure PUBLIC "-//Jetty//Configure//EN"
"http://www.eclipse.org/jetty/configure.dtd">
<Configure class="org.eclipse.jetty.webapp.WebApplicationContext">
<Set name="maxFormContentSize">500000</Set>
</Configure>
```

Migrate Using Imported Settings

Imported settings come from settings you exported from a custom migration. These settings are stored in a file named `migrate.dat` in the `9.8_Software AG_directory \MWS \bin \migrate` directory. Copy the `migrate.dat` file to any directory on machines that host My webMethods Server 9.8 installations to which you want to migrate data.

Migrate Using Imported Settings with Prompting

1. On the 9.8 machine, open a command window or shell, go to the `9.8_Software AG_directory \MWS \bin \migrate` directory, and run the command `migrate.{bat|sh}`.
2. The utility asks for the full path to the old Software AG installation directory or to the Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter Y and, when prompted, provide the full path to the `migrate.dat` file.

Migrate Using Imported Settings without Prompting

On the 9.8 machine, open a command window or shell, go to the `9.8_Software AG_directory \MWS \bin \migrate` directory, and run the command below. If an error occurs, the utility exits.

```
migrate.{bat|sh}
-srcDir full_path_to_old_Software
AG_directory\Zip_file}
-importFile full_path_to_migrate.dat
-silent true
```

8.2 Upgrade: Migrate the My webMethods Server Infrastructure

1. Go to the old `MWS \server \default \bin` directory and open the `server.properties{bat.sh}` file.
2. Migrate your JMX port configurations. Open the `com.softwareag.jmx.connector.pidport.properties` files in the `9.8 SoftwareAG_directory \profiles \MWS_instance_name \configuration \com.softwareag.platform.config.propsloader` directory. If a value in the old `server.properties{bat.sh}` file is different from the corresponding value in the 9.8 file, copy the old value over the 9.8 value.
3. If you customized Java parameters, copy the customizations from the old `server.properties{bat.sh}` file to the 9.8 `custom_wrapper.conf` file. The 9.8 `custom_wrapper` file is located in the `9.8 SoftwareAG_directory \profiles \MWS_instance_name \configuration` directory.

9.0, 9.5, 9.6, or 9.7 Upgrade: Migrate the My webMethods Server Infrastructure

Follow the steps below for each server instance.

1. Migrate your JMX port configurations. Open the `com.softwareag.jmx.connector.pidport.properties` files in the old and 9.8 *SoftwareAG_directory* \profiles\MWS_instance_name \configuration \com.softwareag.platform.config.propsloader directories. If any values in the two files are different, copy the old value over the 9.8 value.
2. If you customized the old My webMethods Server Java Service Wrapper, copy the customizations from the old `wrapper_conf` or `custom_wrapper.conf` files to the 9.8 `custom_wrapper.conf` file. The files are located in the *SoftwareAG_directory* \profiles\MWS_instance_name \configuration directory.

Initialize My webMethods Server

1. If you customized portlet permissions in the old My webMethods Server and want to retain the same permissions after migration, go to the *SoftwareAG_directory* \profiles\MWS_instance_name \configuration directory, open the `custom_wrapper.conf` file, and add this parameter below to the Java Additional Parameters section:

```
wrapper.java.additional.number=-DxmlImport.override.permissions=FALSE
```

2. Initialize My webMethods Server 9.8 by going to the *9.8_SoftwareAG_directory* \MWS\bin directory and running this command:

```
mws.{bat|sh} -s instance_name init
```

The 9.8 components are deployed, and then My webMethods Server shuts down automatically.

3. If you added the parameter in step 1 to the `custom_wrapper.conf` file, remove the parameter from the file.
4. Restart My webMethods Server.

Update Host Names

If the old and new My webMethods Server installations are on different machines, verify the host names for the new installation as follows:

1. Log on to My webMethods Server 9.8 as Administrator and go to the **Administration > My webMethods > Cluster Settings > Advanced Web and Cluster Configuration for MWS** page.

2. If the host name is not correct in the **Host** field for each cluster node and in the **MWS Front End URL** field for all nodes, update the host names.
3. Go to the **Cluster Status and Control** page and restart all cluster nodes.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

7 Completing the Integration Server and Hosted Products Upgrade

■ Before Migrating Data	54
■ Run the Integration Server Migration Utility	54
■ Complete the Integration Server Upgrade	59
■ Complete the ActiveTransfer Upgrade	61
■ Complete the CloudStreams Upgrade	61
■ Complete the Trading Networks Upgrade	62
■ Complete Adapter Upgrades	63
■ Complete eStandards Module Upgrades	64

Before Migrating Data

1. If you are upgrading a Integration Server cluster, read "[Prepare the Old Integration Server](#)" on page 24.
2. If you are upgrading an Integration Server that hosts Deployer, read "[Complete the Deployer Upgrade](#)" on page 80.
3. Make sure Integration Server is shut down.
4. If you are going to migrate packages that you created, and you gave a package a name that starts with Wm, go to the *9.8_Software AG_directory* \IntegrationServer \bin\migrate directory, open the packages.cnf file, and add a <value name></value> tag that identifies that package. If you are going to migrate run-time data for business processes, also add any run-time packages generated from Software AG Designer that you gave a name starting with Wm.

Note: To simplify future upgrades, and as a general best practice, do not give packages names that start with Wm.

Run the Integration Server Migration Utility

The Integration Server migration utility can migrate the following:

- Custom and user-defined packages, configuration files, and, starting in 9.5, custom jar files.
- If the old Integration Server hosted products (for example, Trading Networks) and those products also exist on Integration Server 9.8, the utility migrates data for those products.

9.6 or 9.7 upgrade: If the old Integration Server installation has multiple instances, you run the migration utility once for each instance you want to migrate. You can have the utility migrate an old instance to a 9.8 instance you created after installation or to a new instance the utility creates for you.

For a general description of migration utility behavior, see "[Migration Utilities](#)" on page 31.

Perform a Custom Migration

1. On the 9.8 machine, open a command window or shell and go to the *9.8_Software AG_directory* \IntegrationServer\bin\migrate directory.
2. 8.2 upgrade: If the old and new Integration Server installations are on different operating systems, run the appropriate command below.

When upgrading from...	Run this command...
Windows to UNIX/Linux	<code>export JAVA_TOOL_OPTIONS=-Dfile.encoding=Cp1252</code>
UNIX/Linux to Windows	<code>set JAVA_TOOL_OPTIONS=-Dfile.encoding=UTF-8</code>

3. Run the command `migrate.{bat|sh}`.
4. The utility asks for the full path to one of the following:
 - The old Integration Server installation (for example, `C:\old_Software AG_directory \IntegrationServer` or `/opt/old_Software AG_directory /IntegrationServer`).
 - The Zip file you made earlier.
5. 9.6 or 9.7 upgrade: If the old Integration Server installation has more than one instance, the utility lists the instances and asks which instance to migrate from.
6. The utility asks for the name of the 9.8 instance to which to migrate. If you want to create a 9.8 instance and migrate to it, enter an instance name that does not exist in the 9.8 installation. If you want to migrate to an instance that does exist in the 9.8 installation, enter that instance name.

When you installed Integration Server 9.8, the Software AG Installer copied all packages and language packs you selected during installation into a package repository. If packages exist on the old instance, and the same packages exist in the 9.8 package repository, the utility will install those packages on the 9.8 instance. If you are creating an instance, and language packs exist in the package repository, the utility will install those language packs on the 9.8 instance.

Note: The migration utility can only install the ActiveTransfer Support package on the default Integration Server 9.8 instance.

7. The utility asks whether to import migration settings. Enter N.
8. The utility asks whether to migrate custom packages. If you choose to migrate all custom packages, the utility migrates the following:
 - User-created and business process run-time packages. If the name of any of these packages starts with `Wm`, it will only be migrated if you added the package to the `packages.cnf` file earlier.
 - The `WmChemPayloads`, `WmPapinetPayloads`, `WmRNPIps`, and `WmFINMessages` packages, if present. Each of these packages contains IS documents for its eStandards Module and the schemas for those documents.

If you choose to migrate selected packages only, the utility lists each of the packages listed above and asks whether to migrate it.

9. The utility asks whether to migrate Integration Server configuration files. If you choose to migrate selected configuration files only, the utility lists each configuration file and asks whether to migrate it. There are about 50 configuration files.

If you are using a cloned database with Integration Server 9.8, you will need to respond N to all prompts that list config\jdbc files. If you are using the old database with Integration Server 9.8, you will need to respond Y to these prompts.

10. 9.5, 9.6, or 9.7 upgrade: The utility asks whether to migrate custom jar files. If you choose to migrate selected custom jar files only, the utility lists each custom jar file and asks whether to migrate it.
11. The utility asks you to specify the behavior to use for new Integration Server 9.8 properties and Integration Server 9.8 properties that have new defaults. If you choose to select behavior, the utility lists each property and asks whether to use new behavior or preserve existing behavior.
12. If the Integration Server hosts other products (for example, ActiveTransfer, Mediator, adapters, and eStandards Modules), the migration utility asks whether to migrate the data for those products.

Note: If the Integration Server hosts Deployer, the migration utility will ask whether to migrate Deployer data. Do not do so unless you have read ["Complete the Deployer Upgrade" on page 80](#) and completed all necessary pre-migration tasks.

13. The utility asks whether to migrate Trading Networks information. If you enter Y, the utility asks the following:
 - a. Whether to migrate the Trading Networks configuration file. If you enter Y, the utility copies the configuration properties from the old Trading Networks installation directory, adds them to the new configuration properties in Trading Networks 9.8, and replaces the properties.cnf file in the *9.8_Software AG_directory \IntegrationServer\instances\instance_name \packages\WmTN \config* directory.

Note: If you have a cluster of Trading Networks instances, you must migrate the configuration file for each instance.

- b. 8.2 upgrade: Whether to migrate Trading Networks data. If you enter Y, the utility does the following:
 - Adds an onboarding IDTYPE to the IDTYPE table in the TradingNetworks database component. Trading Networks uses the new onboarding IDTYPE when you create partners using the Trading Networks Partner Onboarding feature.
 - Imports onboarding system template data into the TEMPLATE, TEMPLATEFIELD, TEMPLATEFIELDDEFS, and TEMPLATEFIELDGROUPDEFS tables in the TradingNetworks database component. The data appears in the default questionnaire templates

that accompany invitations you send to new partners using the Partner Onboarding feature.

Note: If you have a cluster of Trading Networks instances, Trading Networks data is shared by all instances in the cluster, so only migrate the data for one instance.

- c. Whether to migrate Trading Networks dashboard data. If you enter Y, the utility purges records from the dashboard tables and populates records from BIZDOC tables into dashboard tables. The dashboard tables include TransactionSummaryData, CustomAttributeVolumeValue, TransactionSuccessFailedData, SuccessFailedChartDocIdMap, and TransactionLateFADData.
14. The utility asks whether to export your settings. If you want to perform other migrations by importing the settings from this session, enter Y. If not, enter N.
15. The utility asks whether to begin migration. If you enter Y, the utility migrates the data you selected. If migration completes successfully, the utility also performs the actions in ["Other Actions Performed by the Migration Utility" on page 58](#).

Migrate Using Imported Settings

Imported settings can come from the following:

- Settings you exported from a custom migration. These settings are stored in a file named migrate.dat in the *9.8_Software AG_directory* \IntegrationServer\bin\migrate directory from which you ran the custom migration. Copy the migrate.dat file to any directory on machines that host Integration Server 9.8 installations to which you want to migrate data.
- Settings in the default migrations provided with Integration Server. For each old release, the settings are stored in a file named migraterelease.sbs.dat file in the *9.8_Software AG_directory* \IntegrationServer\bin\migrate directory. The settings tell the migration utility to migrate the data listed under ["Run the Integration Server Migration Utility" on page 54](#). For Integration Server properties that are new or that have new defaults, the default migration chooses the behavior that best preserves backwards compatibility.

Migrate Using Imported Settings with Prompting

1. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory* \IntegrationServer\bin\migrate directory, and run the command `migrate.{bat|sh}`.
2. The utility asks for the full path to the old Integration Server installation or to the Zip file you made earlier.
3. 9.6 or 9.7 upgrade: If the old Integration Server installation has more than one instance, the utility lists the instances and asks which instance to migrate from.

- The utility asks for the name of the 9.8 instance to which to migrate. If you want to create a 9.8 instance and migrate to it, enter an instance name that does not exist in the 9.8 installation. If you want to migrate to an instance that does exist in the 9.8 installation, enter that instance name.

When you installed Integration Server 9.8, the Software AG Installer copied all packages and language packs you selected during installation into a package repository. If packages exist on the old instance, and the same packages exist in the 9.8 package repository, the utility will install those packages on the 9.8 instance. If you are creating an instance, and language packs exist in the package repository, the utility will install those language packs on the 9.8 instance.

Note: The migration utility can only install the ActiveTransfer Support package on the default Integration Server 9.8 instance.

- The utility asks whether to import migration settings. Enter Y and, when prompted, provide the full path to the migrate.dat file or the migrate{822|901|951|960}sbs.dat file.

Migrate Using Imported Settings without Prompting

On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \IntegrationServer\bin\migrate* directory, and run the command below. The `-instanceName` option specifies both the old instance to migrate from and the 9.8 instance to migrate to; if the 9.8 instance does not exist, the utility will create it. If an error occurs, the utility exits.

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Integration_Server_directory|Zip_file}
-instanceName instance_name
-importFile full_path_to_{migrate.dat|migrate{822|901|951|960|970}sbs.dat}
-silent true
```

Other Actions Performed by the Migration Utility

When migration completes successfully, the migration utility also does the following:

- Deletes old configuration files that are not used by Integration Server 9.8, and deletes old properties that are not used by Integration Server 9.8 from the 9.8 configuration files.
- If the old Integration Server used the embedded database, the utility copies the database tables from the old Integration Server to Integration Server 9.8 and converts the tables to the new format.
- Adds a new property named "Validate schemas using Xerces" to existing Web service descriptors, and sets the new property to the value the `watt.server.wsdl.validateWSDLSchemaUsingXerces` parameter had in the old Integration Server. The new property replaces the functionality provided by that parameter. For information about the new property, see *webMethods Service Development Help 9.8*.

- If the old Integration Server was part of a cluster that used Coherence as its caching software, the utility removes the clustering settings so you can reconfigure the cluster to use a Terracotta-based solution. Support for Coherence was discontinued in release 9.0. For more information, see *webMethods Integration Server Clustering Guide 9.8*.
- 8.2 upgrade: Updates the SoftwareAG-IS-Core.xml caching file to include caches added for Integration Server 9.8.
- If you are using Business Rules and chose to migrate rule data, the utility upgrades Business Rule projects. You might see XML parsing messages due to a Java bug; you can ignore these messages.

Complete the Integration Server Upgrade

Migrate the Flat File Package Properties File

If you used the Flat File package on the old Integration Server, you might have created a properties.cnf file to control some flat file behavior.

8.2, 9.0, or 9.5 upgrade: Go to the old_Software AG_directory\IntegrationServer\instance_name\packages\WmFlatFile\config directory. If the properties.cnf file is present, copy it to the same directory in the 9.8 installation.

9.6 or 9.7 upgrade: Go to the old_Software AG_directory\IntegrationServer\instances\instance_name\packages\WmFlatFile\config directory. If the properties.cnf file is present, copy it to the same directory in the 9.8 installation.

Customize Integration Server Startup Files

The Integration Server startup scripts have been changed in release 9.8. If you customized the old scripts, see *webMethods Integration Server Administrator's Guide* and the Integration Server readme for instructions on how to duplicate those customizations for 9.8.

Update WSDLs

If you have Provider Web services that have an operation with field names starting with xml in the input signature, output signature, header, or faults, do the following:

1. Start Integration Server 9.8.
2. Review the migrationLog.txt file in the 9.8_Software AG_directory\install\logs directory. If you see this error:

```
A property watt.server.xml.ncname.encode.backward.compatibility exists in
config/server.cnf with value as true. Make sure you make the required changes
as specified in the upgrade documentation. Not doing so could have adverse
effects as support for this property may be dropped in a future release.
```

Update your WSDLs as follows:

- a. Open Integration Server Administrator and point to Integration Server 9.8.
- b. Go to the **Settings > Extended** page. If you have the extended setting `watt.server.xml.ncname.encode.backward.compatibility` and it is set to true, reset it to false.
- c. Regenerate the clients for all Provider Web services that have an operation with field names starting with `xml` in the input signature, output signature, header, or faults.

Connect Integration Server Functions to Database Components

Connect Integration Server and the products it hosts to their database components. For instructions, see *Installing webMethods and Intelligent Business Operations Products 9.8*.

Update Host Names

If the old and new Integration Servers are on different machines, update the host names for 9.8 as follows:

1. If you want to resubmit failed business processes in the 9.8 installation, update the host name in the ISCoreAudit database component.
2. Update the host name in the TARGET column in the IS_USER_TASKS database table.
3. Update the SERVERID column in the database tables below.
 - IS_USER_TASKS, WMDOCUMENT, WMERROR, WMRULEDIST, WMSECURITY, WMSESSION, WMTXIN, WMTXOUT, WMTXOUTWMCONTROL
 - WMSERVICE, WMSERVICE4X, WMSERVICEACTIVITYLOG, WMSERVICEASSOC, WMSERVICECUSTOMFLDS

9.0, 9.5, 9.6, or 9.7 Upgrade: Update the Integration ServerNERV Configuration

1. Go to the `9.8_Software AG_directory \profiles\IS_instance_name \configuration \com.softwareag.platform.config.propsloader` directory and open the `com.softwareag.platform.eda.nerv.properties` file. Update JMS provider references to point to your 9.8 JMS provider. Update file system paths to point to the 9.8 installation.
2. Open the same file in the old installation. If any values in the two files other than the path values mentioned above are different, copy the old value over the 9.8 value.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

Complete the ActiveTransfer Upgrade

For instructions and details on all steps below, see *Managing File Transfers with webMethods ActiveTransfer 9.8*.

1. Copy keystore files and the security.cnf file from the old installation to the ActiveTransfer 9.8 installation. Update the keystore file paths.
2. Make sure that port values, host names and IP addresses, and local and remote file paths in event file operations and the virtual file system (VFS) are correct for the ActiveTransfer 9.8 installation.
3. If you used temporary accounts and shared files with the old ActiveTransfer, you can copy that information so you can use it with ActiveTransfer 9.8. Go to the `old_Software AG_directory \IntegrationServer\instances\instance_name` directory for the Integration Server instance that hosts ActiveTransfer and copy the TempAccounts directory to the same location in the 9.8 installation.
4. Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

Complete the CloudStreams Upgrade

If the old Integration Server hosted CloudStreams and you installed the same release or a later release of CloudStreams on the current version of Integration Server, you migrated data and configuration settings for CloudStreams as part of the Integration Server migration. However, you must also complete the tasks below.

1. Set the Integration Server property `watt.server.http.listRequestVars=asNeeded`.
2. Back up any custom packages that contain CloudStreams artifacts.
3. Run the public service `pub.cloudstreams.migration:migrate`, which is available in the `WmCloudStreams` package. The service migrates old CloudStreams artifacts in custom packages that depend on the `WmCloudStreams` package and updates them to be compatible with CloudStreams 9.8. Artifacts include SOAP and REST connector services and connections. The utility logs the results of migration to the Integration Server server log.

The input signature for the service is as follows:

Parameter	Description
<i>allPackages</i>	Boolean string. Optional. Set to: <ul style="list-style-type: none"> ■ <code>true</code> to migrate CloudStreams artifacts from all custom packages that are dependent on WmCloudStreams. ■ <code>false</code> to migrate CloudStreams artifacts from only custom packages specified on the <i>packages</i> parameter. This is the default.
<i>packages</i>	Object. String array containing the names of custom packages to migrate. You must specify at least one package name.

The output signature of the service is a parameter named Result that consists of an array of iData records. The array contains a record for each custom package specified in the input signature. The fields in each record are as follows:

Field	Description
<i>packageName</i>	String. Name of the custom package.
<i>success</i>	String. Value that indicates whether migration succeeded (<code>true</code>) or failed (<code>false</code>).
<i>message</i>	String. Information about the migration, such as number of CloudStreams artifacts found in the package and number of CloudStreams artifacts that were successfully migrated.
<i>info</i>	Object. Optional. If the service issued info messages during the migration, string array of the messages.
<i>errors</i>	Object. Optional. If the service issued errors during the migration, string array of the errors.

4. Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

Complete the Trading Networks Upgrade

If you chose to have the Integration Server migration utility migrate the Trading Networks configuration file, go to the *Software AG_directory* \IntegrationServer \instances*instance_name* \config\caching directory and delete the SoftwareAG-IS-TN.xml file.

Complete Adapter Upgrades

If the old Integration Server hosted adapters and you installed the same release or a later release of those adapters on Integration Server 9.8, you migrated data for those adapters as part of the Integration Server migration. For the adapters below, however, you must also complete the indicated tasks. Then restart Integration Server and complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

webMethods Adapter for JDBC

If the adapter connects to an Oracle RDBMS using an OCI driver, set these environment variables on the Integration Server 9.8 machine:

Platform	Environment Variable Setting
Solaris	LD_LIBRARY_PATH=/ORACLE_HOME/lib
HP	SHLIB_PATH=/ORACLE_HOME/lib
AIX	LIBPATH=/ORACLE_HOME/lib
Linux	LD_LIBRARY_PATH=/ORACLE_HOME/lib

webMethods Package for Microsoft .NET

If you modified the `wmSystem.exe.config` file in the old installation, make the same changes to the same file in the new installation. The file is located in the *Software AG_directory* \IntegrationServer\DotNetHost directory.

webMethods PeopleSoft EnterpriseOne Adapter

If the adapter connects to an Oracle RDBMS, go to the *old_Software AG_directory* \IntegrationServer\lib\jars directory and copy the Oracle JDBC driver file to the *9.8_Software AG_directory* \IntegrationServer\lib\jars\custom directory.

webMethods SAP Adapter

1. Go to the *old_Software AG_directory* \IntegrationServer\lib directory and copy all files whose file name contains the phrase `*sapjco.*` to the *9.8_Software AG_directory* \IntegrationServer\lib\jars\custom directory.

2. Go to the `old_Software AG_directory \IntegrationServer\packages\WmSAP\ns\wm\sap\internal\la\routingListener` directory and copy the `node.ndf` file to the same directory on the 9.8 machine.
3. If the Integration Server 9.8 machine uses UNIX, set these environment variables on the machine:

Platform	Environment Variable Setting
Linux, Solaris	LD_LIBRARY_PATH
HP	SHLIB_PATH
AIX	LIBPATH

4. If the Integration Server 9.8 machine uses Windows, go to the `C:\WINDOWS\system32\drivers\etc` directory on the old Integration Server machine, open the `services` file, and copy all entries that contain the phrase "sapgw" into the same file in the 9.8 installation. Examples of these entries are `sapgw00 3300/tcp`, `sapgw01 3301/tcp`, `sapgw02 3302/tcp`, `sapgw03 3303/tcp`, `sapgw04 3304/tcp`, and `sapgw05 3305/tcp`.

Complete eStandards Module Upgrades

If the old Integration Server hosted eStandards Modules and you installed the same release or a later release of those modules on Integration Server 9.8, you migrated data for those modules as part of the Integration Server migration. For the modules below, however, you must also complete the indicated tasks. Then complete final upgrade tasks as instructed in ["Complete Final Upgrade Tasks for All Products"](#) on page 118.

webMethods FIX Module

If you installed the same release of FIX Module on Integration Server 9.8, go to the `old_Software AG_directory \IntegrationServer\packages` and copy the `WmFIXMessages` package to the `9.8_Software AG_directory \IntegrationServer\instances\instance_name \packages` directory.

webMethods SWIFTNet Module

Reconfigure the module. For instructions, see *webMethods SWIFTNet Module Installation and User's Guide*.

8 Completing the Software AG Designer, Business Process Run-time, and Blaze Upgrades

■ Migrate Software AG Designer Data	66
■ Complete the Business Process Run-time Upgrade	68
■ Update Host Names	69
■ Migrate Closed Loop Analytics	69
■ Complete Final Upgrade Tasks	69
■ Complete the Blaze Rules Upgrade	69

Migrate Software AG DesignerData

Before Migrating Data

1. Open Software AG Designer 9.8 and point to a new 9.8 workspace. For example, you can accept the default workspace98.
2. Install any third-party features you need (for example, support for Subversion). The Eclipse release installed with Software AG Designer 9.8 is Eclipse 4.4, so make sure any features you add are compatible with Eclipse 4.4.
3. If you are upgrading Events, go to the **File > Import** wizard. In the Select panel, go to **Run/Debug > Launch Configurations** and click **Next**. In the **From Directory** field, go to the directory to which you exported your Event Bus Console configurations in [Export Event Bus Console Configurations](#) and select all files with the extension .launch. Click **Finish**.
4. If you exported your preferences before upgrading, do the following:
 - a. Go to the **File > Import** wizard. In the Select panel, go to **General > Preferences** and click **Next**. In the **From preference file** field, specify the .epf file to which you exported your preferences. Click **Finish**.
 - b. Go to **Window > Preferences**. In the preferences window, go to the **Java > Installed JREs > Execution Environments** page, if multiple JREs are listed, make sure JRE 1.8 is selected as the default, or remove older JREs from the list. Also update settings that point to old product installations to point to 9.8 product installations instead. For example, for My webMethods Server, update the **Server > Runtime Environments** settings.

Migrate Business Process and Business Rule Projects

1. Start Software AG Designer 9.8.
2. If your projects are stored in a source control system, use the import wizard for that system (for example, **Import SVN > Project from SVN**). If your projects are stored in the Software AG Designer workspace, do the following:
 - a. Go to the **File > Import** wizard.
 - b. In the Select panel, go to **General > Existing Projects into Workspace** and click **Next**.
 - c. In the Import Projects panel, do the following:
 - a. Click **Select Root Directory** and go to the workspace that contains a type of project you want to import, or click **Select Archive File** and go to the directory that contains a type of project you want to import.
 - b. In the **Projects** box, select the projects to import. Select **Copy projects into workspace**. Click **Finish**.

3. Verify the following:
 - All imported business process projects appear in the **Solutions** tab under the Processes node.
 - All imported business rules projects appear in the **Solutions** tab under the Rules node.
4. If you imported business rule projects, right-click each project in the Rules Explorer and then click **Upgrade Project**.

Migrate Task Application Projects

1. In Software AG Designer 9.8, go to **Window > Preferences > Server > Runtime Environments**. If the **Installed server runtimes** list does not include a My webMethods Server 9.8, add one.
2. Go to **Software AG > Task Development**. In the preferences window, make sure all your other task-related Software AG Designer preferences are correct, and then click **OK**.
3. Go to the **File > Import** wizard.
 - a. In the Select panel, go to **Software AG > Existing CAF Projects into Workspace** and then click **Next**.
 - b. In the Import Projects panel, click **Select Root Directory** and go to the workspace or source control system that contains your task application projects, or click **Select Archive File** and go to the directory that contains your task application projects. In the **Projects** box, select the projects to import.
 - c. Select **Copy projects into workspace** and then click **Finish**.
4. Make sure all imported task application projects appear in the **Solutions** tab, under the Tasks node.
5. If errors appear in the **Problems** tab, click the **Navigator** tab. Right-click each task application project, click **CAF Tools**, and click **Repair CAF Project**. If errors still appear, restart Software AG Designer.
6. Publish the migrated task application projects to My webMethods Server 9.8. For instructions, see *webMethods BPM Task Development Help 9.8*.

Migrate CloudStreams Server Definitions and Projects

1. Start Software AG Designer 9.8.
2. Import CloudStreams Server definitions as follows:
 - a. Go to the **CloudStreams Development** perspective, then go to **Window > Preferences**. On the **Preferences** dialog box, in the left navigation bar, go to **Software AG > CloudStreams Servers**.

- b. Click **Import**, select the .properties file you exported, and click **Open**. Software AG Designer asks whether to overwrite existing servers. Click **OK**, and then click **OK** again to close the **Preferences** dialog box.
3. Import CloudStreams Governance projects as follows:
 - a. Go to the **File > Import** wizard. In the Select panel, go to **Software AG > CloudStreams Governance Project** and then click **Next**.
 - b. In the Import CloudStreams Governance Project panel, specify the directory that contains the projects you exported in the **Select Root Directory** field. In the **Projects** box, select the projects to import.
 - c. Select **Copy projects into workspace** and then click **Finish**.
 - d. In the **CloudStreams Development** perspective, make sure all imported governance projects appear in the **CloudStreams Governance** tab.
4. Repeat the previous step to import CloudStreams Provider projects.

Migrate Integration Server Definitions

1. In Software AG Designer 9.8, go to **Window > Preferences**. On the **Preferences** dialog box, in the left navigation bar, go to **Software AG > Integration Servers**.
2. Click **Import**, select the .properties file you exported, and click **Open**. Software AG Designer asks whether to overwrite existing servers. Click **OK**, and then click **OK** again to close the **Preferences** dialog box.

Complete the Business Process Run-time Upgrade

Note: If you have running process instances that are based on a process model created before you upgraded to 9.8, and you now want to regenerate that process model in Software AG Designer 9.8, change the version number of the process model to the next sequence number before regenerating it. If you do not do so, the process instances will not behave as expected. When you enable the new version of the process model for execution in My webMethods Server, you will be asked if you want to upgrade running processes; respond **No**.

1. If you upgraded webMethods Broker, make sure the new Broker that will be used by the Process Engines has the same name in 9.6 that it had in the old release. To do so, open Integration Server Administrator and point to Integration Server 9.8. Go to the **IS Settings > Messaging > webMethods Messaging Settings** page and check the Broker name.
2. If you migrated process run-time packages (see ["Run the Integration Server Migration Utility" on page 54](#)), make sure the packages exist in the `9.8_Software AG_directory \IntegrationServer \instances \instance_name \packages` directory. The

package names are the project names or custom names you specified in Software AG Designer.

3. Open Integration Server Administrator and point to Integration Server 9.8. Go to the **Settings > JDBC Pools** page and connect the ProcessEngine and ProcessAudit functions to their database components. For instructions, see *Installing webMethods and Intelligent Business Operations Products 9.8*.
4. Make sure document retrieval for all webMethods messaging triggers is enabled on Integration Server 9.8. For instructions, see *webMethods Integration Server Administrator's Guide 9.8*.

Update Host Names

If the old and new business process product installations are on different machines, update the host name in the SERVERID column in the database tables below.

- PRTINSTANCEITER, PRTPROCESS, PRTPROCLOCK, PRTQUEUE, PRTSTEPLOCK, PRTSTEPSTATE, PRTSTEPSUSPENSION
- PRA_ERROR
- PRA_PROCESS, PRA_PROCESS_AT_REST, PRA_PROCESS_CUSTOM, PRA_PROCESS_STEP, PRA_PROCESS_STEP_LOOP
- PRA_STEP_LOGGED_FIELD, PRA_STEP_LOOP_LOGGED_FIELD, PRA_STEP_MESSAGE, PRA_STEP_TRANSITION
- WMSTG_PROCESS_CONTROL, WMSTG_PROCESS_STEP, WMRULEDIST

Migrate Closed Loop Analytics

Migrate all assets involved in Closed Loop Analytics by migrating business process, business rule, and task application projects as described in this chapter.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

Complete the Blaze Rules Upgrade

Migrate Blaze rules using the instructions in *Using Blaze Rules with BPM and CAF9.8*.

9 Completing the Optimize Upgrade

■ Configure and Start the Terracotta Server Array	72
■ Update Connection to Other Products	72
■ Complete the Infrastructure Data Collector Upgrade	73
■ Deploy the Optimize Environment	75
■ Switch from webMethods Broker to Universal Messaging	77
■ Complete Final Upgrade Tasks	78


Configure and Start the Terracotta Server Array

If you are clustering Analytic Engines in release 9.8, you must configure and start the Terracotta Server Array. For instructions, see *Getting Started with the webMethods Product Suite and Terracotta 9.8*, *Configuring BAM 9.8*, and the Terracotta BigMemory Max 4.1 or later documentation.

Update Connection to Other Products

Update Connection to Process Engine

If you are using Optimize for Process, do the following:


1. Open Integration Server Administrator and point to an Integration Server 9.8 that hosts a Process Engine.
2. Go to the **Packages > Management** page and click  for the WmPRT package.
3. Click **Settings** in the left navigation bar and then click **Edit Process Engine Settings**.
4. In the **JMS Server URL** field, identify the JMS provider host name and port as follows:

JMS Provider	URL Format
webMethods Broker	broker://host:port/Broker_name . The default port is 6849.
Universal Messaging	nsp://host:port . The default port is 9000.

5. Click **Submit** and then reload the WmPRT package.
6. Repeat these steps for every Integration Server that hosts a Process Engine.

Update Connection to Optimize Support Package

If you are using Optimize built-in services, do the following:

1. Open Integration Server Administrator and point to the Integration Server 9.8 that hosts an Optimize Support package.
2. Go to the **Packages > Management** page and click  for the WmOptimize package.
3. Identify the Analytic Engine host machine and port. The default port is 12503.
4. In the **JMS Server URL** field, identify the JMS provider host name and port as follows:

JMS Provider	URL Format
webMethods Broker	broker://host:port/Broker_name . The default port is 6849.
Universal Messaging	nsp://host:port . The default port is 9000.

- Click **Submit** and then reload the WmOptimize package.
- Repeat these steps for every Integration Server that hosts an Optimize Support package.

Complete the Infrastructure Data Collector Upgrade

8.2 Upgrade: Restart Monitoring of Clustered Integration Servers

If you had Integration Servers that were clustered using Oracle Coherence, you reconfigured them earlier to use a Terracotta-based clustering solution instead. You can now rediscover the clustered Integration Servers you want to monitor, re-select the KPIs you noted in ["Prepare the Old Optimize"](#) on page 26, and restart monitoring. For instructions, see *Administering webMethods Optimize 9.8*.

9.7 Upgrade: Start Monitoring 4.1 or Later Terracotta Server Arrays

For instructions on each step, see *Administering webMethods Optimize 9.8*.

- Go to **Applications > Administration > Infrastructure Components > Discovery**.
- Delete all Terracotta (Asset) entries.
- Rediscover 4.1 or later Terracotta Server Arrays you want to monitor.
- Go to **Applications > Administration > Infrastructure Components > Monitored Components**. Select Terracotta Server Array monitors, components, and KPIs.

Migrate Customizations to the Java Service Wrapper

If you customized the old Infrastructure Data Collector Java Service Wrapper, copy the customizations from the old wrapper_conf or custom_wrapper.conf files to the 9.8 custom_wrapper.conf file. The files are located in the *Software AG_directory \profiles \InfraDC \configuration* directory.

Run the Infrastructure Data Collector Migration Utility

The Infrastructure Data Collector utility migrates asset configuration files and the JAAS configuration for Infrastructure Data Collector.

For a general description of migration utility behavior, see ["Migration Utilities" on page 31](#).

Before Migrating Data

1. Make sure the old Infrastructure Data Collector is shut down.
2. Go to the *9.8_Software_AG_directory* `infrastructuredc\bin\migrate\release` directory and open the `migrateConfigFile.properties` file. Identify each asset configuration file or directory that contains asset configuration files to migrate using the format shown in the `migrateConfigFile.properties` file.

Perform a Custom Migration

1. On the 9.8 machine, open a command window or shell, go to the *9.8_Software_AG_directory* `infrastructuredc\bin\migrate` directory and run the command `migrate.{bat|sh}`.
2. The utility asks for the full path to one of the following:
 - The old Infrastructure Data Collector installation (for example, C:
`\old_Software_AG_directory\infrastructuredc` or `/opt/old_Software_AG_directory/infrastructuredc`).
 - The Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter N.
4. The utility asks whether to migrate SNMP asset configuration files. The migration utility will migrate the old files you identified in the `migrateConfigFile.properties` file in ["Before Migrating Data" on page 74](#).
5. The utility asks whether to migrate the JAAS configuration for Infrastructure Data Collector. If you enter Y, the utility asks whether to replace references to the old directory in the JAAS configuration file with references to the 9.8 directory.
6. The utility asks whether to export your settings. If you want to perform other migrations by importing the settings from this session, enter Y. If not, enter N.
7. The utility asks whether to begin migration. If you enter Y, the utility migrates the data you selected.

If you chose to migrate asset configuration files, and a file that has the same name as an old file already exists in the Infrastructure Data Collector 9.8 installation, the utility backs up the 9.8 file to the *9.8_Software_AG_directory* `infrastructuredc\migrationbackup` directory before migrating the old file. The same is true for the JAAS configuration file.

Migrate Using Imported Settings

Imported settings can come from the following:

- Settings you exported from a custom migration. These settings are stored in a file named `migrate.dat` in the `9.8_Software AG_directory \infrastructuredc\bin\migrate` directory from which you ran the custom migration. Copy the `migrate.dat` file to any directory on machines that host Infrastructure Data Collector 9.8 installations to which you want to migrate data.
- Settings in the default migrations provided with Infrastructure Data Collector. For each old release, the settings are stored in a file named `migraterelase sbs.dat` file in the `9.8_Software AG_directory \infrastructuredc\bin\migrate` directory. The settings tell the migration utility to migrate the data listed in ["Complete the Infrastructure Data Collector Upgrade" on page 73](#).

Migrate Using Imported Settings with Prompting

1. On the 9.8 machine, open a command window or shell, go to the `9.8_Software AG_directory \infrastructuredc\bin\migrate` directory, and run the command `migrate.{bat|sh}`.
2. The utility asks for the full path to the old Infrastructure Data Collector installation or to the Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter Y and, when prompted, provide the full path to the `migrate.dat` file or the `migrate{822|901|951|960|970}sbs.dat` file.

Migrate Using Imported Settings without Prompting

On the 9.8 machine, open a command window or shell, go to the `9.8_Software AG_directory \infrastructuredc\bin\migrate` directory, and run the command below. If an error occurs, the utility exits.

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Infrastructure Data
Collector_directory|Zip_file}
-importFile full_path_to_{migrate.dat|{821|901|951|960|970}sbs.dat}
-silent true
```

Deploy the Optimize Environment

Unless otherwise noted, see *Configuring BAM 9.8* for detailed instructions on the Optimize steps in the sections below.

Not Reuse the Old Optimize CCS Environment Definition

If you do not want to reuse your old Central Configuration System (CCS) Environment definition, start My webMethods Server 9.8, open My webMethods, and configure your Optimize 9.8 environment.

For Infrastructure Data Collector, when you define hosts, enter the same value in the **Host Name or IP Address** field of the Add/Edit Host dialog box that you had in the old release.



Note: If the 9.8 value is not identical to the old value, you will have to stop monitoring existing assets, rediscover them, and reselect them for monitoring after you finish upgrading.

Skip to the step for deploying the Optimize 9.8 environment, below.

Reuse the Old Optimize CCS Environment Definition

To reuse the old CCS environment definition that you exported to file before installing Optimize 9.8, follow the steps below.

1. Start My webMethods Server 9.8 and open My webMethods.
2. Import the old CCS environment, as follows:
 - a. Go to the **Applications > Administration > System-Wide > Environments > Define Environments** page.
 - b. Click **Import Environment**. On the Import Environments dialog box, click **Browse**, go to the file to which you exported the old CCS environment, and click **OK**. When prompted, confirm the migration.
 - c. When asked "Do you wish to migrate the environment to the latest version?", click **Yes**. The CCS imports the old environment and migrates it to Optimize 9.8 Logical Servers.
3. Update the database pool connection settings for the database pool definition associated with the environment you imported, as follows:
 - a. Go to the **Applications > Administration > System-Wide > Environments > Database Pool Configuration** page.
 - b. In the **Pool** column, click the imported database pool.
 - c. In the **Pool Settings** area, set the **Minimum Connections** field to 8, and the **Maximum Connections** field to 60.
 - d. Make sure the database connection information is correct. In the **Database Connection** area, click **Test** and make sure the message `Test Passed` displays at the bottom of the page.
 - e. Click **Save**.
4. Go to the **Applications > Administration > System-Wide > Environments > Define Environments** page. Under **Environment Name**, click the environment you imported.
5. If you are using a different JMS provider with the 9.8 release than you used with the old release, do the following:

- a. Click the **Configure Servers** tab. Under **Configuration**, open the **Default Settings** tree and click **JNDI Configuration**.
 - b. In the **Naming Factory Type** list, click the JMS provider you are using with the 9.8 release. If you are using webMethods Broker, complete the **Broker Name** field. If you are using Universal Messaging, clear the field.
 - c. Click the **Map Endpoints** tab. In the first **Incoming Connections** field for the JMS Provider, type `broker` if you are using webMethods Broker or `nsp` if you are using Universal Messaging. In the second field, type the port number.
 - d. Click **Save**.
6. Validate the environment as follows:
 - a. Click each tab that shows the icon  except the **Validate** tab. If you change any settings on a tab, and the tab has a **Save** button, click **Save**.
 - b. When all tabs except the **Validate** tab show , click the **Validate** tab.
 7. Click **Finish**.

Deploy the Optimize 9.8 Environment

1. Start the Optimize 9.8 components you installed. These components can include the Analytic Engine, Infrastructure Data Collector, and the Web Services Data Collector.
2. Start the JMS provider you are using with the 9.8 release.
3. Deploy your Optimize 9.8 environment. For instructions, see *Configuring BAM 9.8*.
4. In My webMethods, go to the **My webMethods > System Settings > Servers** page.
 - If you have BPMS installed, select the **BPM and BAM** option and identify the upgraded Analytic Engine and the Integration Server that hosts Monitor. You can identify the hosts using DNS name or IP address.
 - If you have BPMS installed, but are not using BPM (that is, you are using only Optimize), select the **BAM only** option and identify the upgraded Analytic Engine. You can identify the host using DNS name or IP address.

Click **Check Server Status** and make sure the server is available (green icon). It might take some time for the server to become available (for example, 15 minutes). Then click **Save**.

Switch from webMethods Broker to Universal Messaging

If you were using webMethods Broker as your JMS provider in the old release, and are using Universal Messaging in 9.8, do the following:

1. Start Universal Messaging and open Universal Messaging Enterprise Manager.

2. Connect to a server instance, then click it.
3. Click the **Config** tab. Expand **Global Values** and set AllowRealmAdminFullAccess to true.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

10 Completing the Asset Build Environment and Deployer Upgrades

■ Complete the Asset Build Environment Upgrade	80
■ Complete the Deployer Upgrade	80

Complete the Asset Build Environment Upgrade

Go to the *old_Software AG_directory \common \AssetBuildEnvironment \master_build* directory and open the *build.properties* file. Copy the property values to the same file in the 9.8 installation.

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

Complete the Deployer Upgrade

You have two options for migrating Deployer data:

- You can migrate all Deployer global (default) settings, server aliases, target groups, and projects using the Integration Server migration utility.
- You can choose whether to migrate global (default) settings, and you can choose which server aliases, target groups, and projects to migrate using the Deployer migration feature.

The instructions for each option are provided below.

For instructions on tasks you must perform in Deployer, see *webMethods Deployer User's Guide*.

Important: Migrating Deployer projects does not convert the assets in those projects from their old release to 9.8. You must migrate all product assets as instructed earlier in this guide before migrating Deployer settings, server aliases, target groups, and projects.

Before Migrating Deployer Settings, Server Aliases, Target Groups, and Projects

1. Make sure you have upgraded all source and target product servers that were defined in the old Deployer to their 9.8 releases, and have migrated all product data and assets to the 9.8 releases, as instructed in this guide.
2. Make sure you have upgraded the Integration Server 9.8 that hosts Deployer 9.8 and migrated its data using the instructions in this guide.
3. Make sure you are a member of the Administrators ACL, or of the Internal, Developer ACLs, and DeployerAdmin ACLs, for both the Integration Server that hosts the old Deployer and the Integration Server that hosts Deployer 9.8. For instructions, see *webMethods Integration Server Administrator's Guide 9.8*.

4. In the old Deployer, edit all connections to old source and target product servers to point to the new 9.8 source and target product servers. You do not have to supply user names and passwords if you do not know them.
5. If you installed Deployer 9.8 on the same machine as the old Deployer, you might want Deployer 9.8 to use the ports that were used by the old Deployer. If so, do the following:
 - a. In Integration Server Administrator, point to the Integration Server that hosts the old Deployer and go to the **Security > Ports** page.
 - b. Click **Add Port**, click **webMethods/HTTP**, and click **Submit**. Enter the port details, click **Yes** for **Enable**, click **WmRoot** under the package name, and click **Save Changes**.
 - c. Click **Change Primary Port**, click the new port, and click **Update**.
 - d. In the **Access Mode** column of the new port, click **Edit**. In the **Edit Access Mode** page for the port, click **Set Access Mode to Allow by Default**. Click **OK** on the warning pop-up. The message `Successfully changed access mode for Port Service Access Settings` appears.
 - e. Disable the old port by clicking **Yes** in the **Enabled** column for that port.

Migrate All Deployer Settings, Server Aliases, Target Groups, and Projects Using the Integration Server Migration Utility

In the previous section, you upgraded the Integration Server that hosts Deployer 9.8 and migrated its data. Do not start the host Integration Server before you migrate Deployer settings, server aliases, target groups, and projects. If you do, a file named `MIGRATION_DONE` is created in the `WmDeployer` package, and you will not be able to migrate Deployer settings, server aliases, target groups, and projects. If you did start the host Integration Server, delete this file.

Migrate Deployer settings, server aliases, target groups, and projects using the instructions in ["Completing the Integration Server and Hosted Products Upgrade" on page 53](#).

Migrate Deployer Settings, Server Aliases, Target Groups, and Projects Using the Deployer Migration Feature

Start the Old Deployer and Deployer 9.8 and Migrate Global (Default) Settings

1. Start the old Deployer and Deployer 9.8.
2. In Deployer 9.8, go to the **Tools > Migrate Data** page. In the **Host** and **Port** fields, identify the old Deployer to migrate from. In the **User** and **Password** fields, provide the user name and password to use to log onto the old Deployer.

3. Deployer 9.8 lists the server aliases, target groups, or projects to migrate in the right-hand pane. In the **Page Size** list, click the number of server aliases or projects to list at a time.
4. In Deployer, you set defaults for all projects, such as defaults for dependency checking, project locking, and general deployment. You also set defaults for Integration Server and Trading Networks projects, such as defaults for suspending assets during deployment, overwriting existing assets, and activating assets after deployment. If you want to migrate these default settings, in Deployer 9.8, click **Migrate Default Settings**.

Migrate Server Aliases (Connection Information)


You can choose the server aliases to migrate. However, if you want to migrate a target group, you must either migrate all server aliases in that target group, or remove the server aliases you do not want to migrate from the target group.

1. In Deployer 9.8, click **Migrate Server Aliases**. In the **Select Server** list, click the type of server whose aliases to migrate.
2. In the **Select** column, select the check boxes next to the server aliases you want to migrate. In the **Point Selected Aliases to Servers** list, click **9.8**.
3. Click **Migrate Server Aliases**. Deployer migrates the selected server aliases, then disables those rows and displays **Yes** in the **Migrate** column for the server aliases.
4. Repeat the steps above for each type of server whose aliases to migrate.
5. You cannot migrate Integration Server HTTPS server aliases from Deployer. Manually recreate these server aliases in the host Integration Server.

Migrate Target Groups

1. If you did not migrate all server aliases in the target groups you want to migrate, in the old Deployer, remove those server aliases from the target groups.


Important: If target groups include invalid server aliases, deployment of projects that use those target groups fails.

2. In Deployer 9.8, click **Migrate Target Groups**. In the **Select Server** list, click the type of server whose target groups to migrate.
3. In the **Simulate Migration** column, click  in the row for each target group to migrate.
Deployer displays a message that indicates whether the server aliases that are referenced by the target group have been migrated. If any server aliases have not been migrated, migrate them or remove them from the indicated target group.
4. In the **Select** column, select the check boxes next to target groups to migrate. In the **Point Selected Aliases to Groups** list, click **9.8**.
5. Click **Migrate Target Groups**. Deployer migrates the selected target groups, then disables those rows and displays **Yes** in the **Migrate** column for the groups.

Migrate Deployment Projects

1. Start all 9.8 source product servers referenced by Deployer projects.

Note: If a server referenced by a Deployer project is not running, or the connection settings for a server referenced by a project are not correct, you will have to rebuild the project later. Deployer will display messages to this effect.

2. In Deployer 9.8, click **Migrate Projects**.
3. In the right-hand pane, in the **Simulate Migration** column, click  row for each project to migrate. Deployer displays a message that indicates whether the source servers that are referenced by the project are running. If the message indicates that any of the referenced source servers are not running, start those servers.
4. In the **Select** column, select the check boxes next to the projects to migrate, and then click **Migrate Projects**. Deployer displays a new window with a dynamic migration report. You can display the migration report for each project in the **Report** column in the original window. You can display the latest migration report by clicking **View Latest Migration Report**.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

11 Completing the Command Central Upgrade

■ Before Migrating Data	86
■ Migrate the Command Central Infrastructure	86
■ Run the Command Central Migration Utility	87
■ Complete the Command Central Upgrade	89

Before Migrating Data

Make sure the old Command Central is shut down.

Migrate the Command Central Infrastructure

1. If you customized the Command Central Java Service Wrappers, copy the customizations from the old wrapper_conf or custom_wrapper.conf files to the 9.8 custom_wrapper.conf file. The files are located in the *Software AG_directory* \profiles\CCE\configuration directory.
2. For each case below, open the file in the indicated directory in the old installation and in the 9.8 installation. If a value in the old file is different from the corresponding value in the 9.8 file, copy the old value over the 9.8 value. In addition, update file system paths in the 9.8 files to point to the 9.8 installation.

Configuration	File Name in Old and 9.8 Software AG Directories
Debug	\profiles\CCE\configuration \com.softwareag.platform.config.propsloader \com.softwareag.platform.debug.properties
JMX ports	\com.softwareag.jmx.connector.pid- <i>port</i> .properties
Security authorization	\com.softwareag.security.authz.pid.properties
Security LDAP	\com.softwareag.security.ldap.pid.properties
Single sign-on service	\com.softwareag.sso.pid.properties
SSH	\com.softwareag.sshd.pid.properties
Watchdog	\com.softwareag.config.watchdog.pid.properties
Journal logger	\profiles\CCE\configuration\logging\log_config.xml
9.0 or 9.5 upgrade:	\profiles\CCE\configuration\security\passman \scPassMan.config.xml

Configuration	File Name in Old and 9.8 Software AG Directories
Password manager	

3. 9.0 or 9.5 upgrade: For each type of configuration below, copy the file from the indicated directory in the old installation to the same directory in the 9.8 installation.

Configuration	File Name in Old and 9.8 Software AG Directories
JAAS	\profiles\CCE\configuration\jaas.config file
Password manager	\profiles\CCE\configuration\security\passman\defaultPassStore.dat and mpw.dat

4. For each case below, open the file in the indicated directory and update file system paths to point to the 9.8 installation.

Configuration	File Name in 9.8 Software AG Directories
JAAS	\profiles\CCE\configuration\jaas.config
Password manager	\profiles\CCE\configuration\security\passman\defaultPassStore.dat and mpw.dat

Run the Command Central Migration Utility

The Command Central migration utility can migrate the following:

- Environment configurations, which include metadata about the environment, nodes (installations), and products that Command Central is managing.
- Users, roles, and groups.
- Port configurations.
- 9.5, 9.6, or 9.7 upgrade: License tools and templates.
- 9.6 or 9.7 upgrade: JAAS configuration and password manager data.
- 9.7 upgrade: ehCache DiskStore.

For a general description of migration utility behavior, see "[Migration Utilities](#)" on page 31.

Perform a Custom Migration

1. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \CommandCentral\migrate\bin* directory, and run the command `migrate.{bat|}`.
2. The utility asks for the full path to one of the following:
 - The Software AG directory that contains the old Command Central installation (for example, *C:\old_Software AG_directory* or */opt/old_Software AG_directory*).
 - The Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter N.
4. The utility asks whether to migrate environment configuration data and users, roles, and groups.
5. 9.5, 9.6, or 9.7 upgrade: The utility asks whether to migrate license tools and templates.
6. 9.6 or 9.7 upgrade: The utility asks whether to migrate the JAAS configuration and password manager data.
7. 9.7 upgrade: The utility asks whether to migrate ehCache DiskStore data.
8. The utility asks whether to migrate HTTP and HTTPS port configurations.
9. The utility asks whether to export your settings. If you want to perform other migrations by importing the settings from this session, enter Y. If not, enter N.
10. The utility asks whether to begin migration. If you enter Y, the utility migrates the data you selected.

Migrate Using Imported Settings

Imported settings can come from the following:

- Settings you exported from a custom migration. These settings are stored in a file named `migrate.dat`. The file is stored in the *9.8_Software AG_directory \CCE\migrate\bin* directory from which you ran the custom migration. Copy the `migrate.dat` file to any directory on machines that host Command Central 9.8 installations to which you want to migrate data.
- Settings in the default migrations provided with Command Central. For each old release, the settings are stored in a file named `migraterelase sbs.dat` file in the *9.8_Software AG_directory \CCE\migrate\bin* directory. The settings tell the migration utility to migrate the data listed under "[Run the Command Central Migration Utility](#)" on page 87.

Migrate Using Imported Settings with Prompting

1. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \CCE\migrate\bin* directory, and run the command `migrate.{bat|}`.
2. The utility asks for the full path to the Software AG directory that contains the old Command Central installation or to the Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter Y and, when prompted, provide the full path to the `migrate.dat` file or the `migrate{901|951|960}sbs.dat` file.

Migrate Using Imported Settings without Prompting

On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \CCE\migrate\bin* directory, and run the command below. If an error occurs, the utility exits.

```
migrate.{bat|}
{-srcDir|-srcFile} full_path_to_{old_Software_AG_directory|Zip_file}
-importFile full_path_to_{migrate.dat|migrate{901|951|960}sbs.dat}
-silent true
```

Complete the Command Central Upgrade

9.6 or 9.7 Upgrade: Restore Trusted Authentication for JAAS

If you chose to migrate your JAAS configuration, follow the steps below to restore trusted authentication.

1. Go to the *9.8_Software AG_directory \profiles\CCE\configuration\security* directory and open the `/jaas.config` file.
2. Add the entries below to the Default login context:

```
com.softwareag.security.idp.saml.lm.JMXDelegatedAuthLoginModule sufficient;
com.softwareag.security.idp.saml.lm.SAMLAssertIssuerLoginModule optional;
```

Update the Command CentralNERV Configuration

Go to the *9.8_Software AG_directory \profiles\CCE\configuration \com.softwareag.platform.config.propsloader* directory and open the `com.softwareag.platform.eda.nerv.properties` file. Update JMS provider references to point to your 9.8 JMS provider, if they do not already. Update file system paths to point to the 9.8 installation.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in ["Complete Final Upgrade Tasks for All Products" on page 118](#).

12 Complete the Platform Manager Upgrade

- 9.0 or 9.5 Upgrade: Complete the Platform Manager Upgrade 92
- 9.6 or 9.7 Upgrade: Complete the Platform Manager Upgrade 94

9.0 or 9.5 Upgrade: Complete the Platform Manager Upgrade

Before Migrating Data

Make sure the old Platform Manager is shut down.

Migrate the Platform Manager Infrastructure

1. Migrate HTTP port configurations as follows:
 - a. Go to the *old_Software AG_directory* \profiles\SPM\configuration \com.softwareag.platform.config.propsloader directory. If you see a single com.softwareag.catalina.connector.http.pid-*port*.properties file, open that file. If you see multiple files with that name, open the file that specifies the port you want to designate as the default port in 9.8.

Go to the same directory in the 9.8 installation and open the com.softwareag.catalina.connector.http.pid-*port*.properties file. If any values in the two files are different, copy the old value over the 9.8 value.
 - b. Check for additional com.softwareag.catalina.connector.http.pid-*port*.properties files for HTTP ports in the above-mentioned old directory. Copy all such files to the 9.8 directory and then add these elements to each copied file:
 - `alias=unique_identifier`. This Software AG-specific attribute uniquely identifies each connector when you view its configuration in the Platform Manager. The alias for the default HTTP connector you set up above is defaultHttp. You could use that alias as a base, and add a unique suffix to create the alias for each additional connector (`alias=defaultHttp2`, `alias=defaultHttp3`, and so on).
 - `enabled=true`. This Software AG-specific attribute and value are required for this connector to start.
2. Repeat the previous step to migrate HTTPS port configurations, but use the com.softwareag.catalina.connector.https.pid-*port*.properties file, and add `alias=defaultHttps` to the default HTTPS connector.
3. If you customized the Platform Manager Java Service Wrapper, copy the customizations from the old wrapper_conf or custom_wrapper.conf files to the 9.8 custom_wrapper.conf file. The files are located in the *Software AG_directory* \profiles \SPM\configuration directory.
4. For each case below, open the file in the indicated directory in the old installation and in the 9.8 installation. If a value in the old file is different from the corresponding value in the 9.8 file, copy the old value over the 9.8 value. In addition, update file system paths in the 9.8 files to point to the 9.8 installation.

Configuration	File Name in Old and 9.8 Software AG Directories
Debug	\profiles\SPM\configuration\com.softwareag.platform.config.propsloadcom.softwareag.platform.debug.properties
JMX ports	\com.softwareag.jmx.connector.pid- <i>port</i> .properties
Security authorization	\com.softwareag.security.authz.pid.properties
Security LDAP	\com.softwareag.security.ldap.pid.properties
Single sign-on service	\com.softwareag.sso.pid.properties
SSH	\com.softwareag.sshd.pid.properties
Watchdog	\com.softwareag.config.watchdog.pid.properties
Journal logger	\profiles\SPM\configuration\logging\log_config.xml
Password manager	\profiles\CCE\configuration\security\passman\scPassMan.config.xml

- Go to the *old_Software AG_directory* \common\conf directory. Copy the user.txt, roles.txt, and groups.txt files to the *9.8_Software AG_directory* \profiles\SPM\configuration\security directory.
- For each type of configuration below, copy the file from the indicated directory in the old installation to the same directory in the 9.8 installation. Then open the copied (9.8) files and update file system paths to point to the 9.8 installation.

Configuration	File Name in Old and 9.8 Software AG Directories
JAAS	\profiles\SPM\configuration\jaas.config file
Password manager	\profiles\SPM\configuration\security\passman\defaultPassStore.dat and mpw.dat

Update the Platform Manager NERV Configuration

Go to the *9.8_Software AG_directory* \profiles\SPM\configuration \com.softwareag.platform.config.propsloader directory and open the com.softwareag.platform.eda.nerv.properties file. Update JMS provider references to point to your 9.8 JMS provider, if they do not already. Update file system paths to point to the 9.8 installation.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

9.6 or 9.7 Upgrade: Complete the Platform Manager Upgrade

Before Migrating Data

Make sure the old Platform Manager is shut down.

Migrate the Platform Manager Infrastructure

1. If you customized the Platform Manager Java Service Wrapper, copy the customizations from the old wrapper_conf or custom_wrapper.conf files to the 9.8 custom_wrapper.conf file. The files are located in the *Software AG_directory* \profiles \SPM\configuration directory.
2. For each case below, open the file in the indicated directory in the old installation and in the 9.8 installation. If a value in the old file is different from the corresponding value in the 9.8 file, copy the old value over the 9.8 value. In addition, update file system paths in the 9.8 files to point to the 9.8 installation.

Configuration	File Name in Old and 9.8 Software AG Directories
Debug	\profiles\SPM\configuration\com.softwareag.platform.config.propsloadecom.softwareag.platform.debug.properties
JMX ports	\com.softwareag.jmx.connector.pid- <i>port</i> .properties
Security authorization	\com.softwareag.security.authz.pid.properties

Configuration	File Name in Old and 9.8 Software AG Directories
Security LDAP	\com.softwareag.security.ldap.pid.properties
Single sign-on service	\com.softwareag.sso.pid.properties
SSH	\com.softwareag.sshd.pid.properties
Watchdog	\com.softwareag.config.watchdog.pid.properties
Journal logger	\profiles\SPM\configuration\logging\log_config.xml

- For each case below, open the file in the indicated directory and update file system paths to point to the 9.8 installation.

Configuration	File Name in 9.8 Software AG Directories
JAAS	\profiles\SPM\configuration\jaas.config
Password manager	\profiles\SPM\configuration\security\passman \defaultPassStore.dat and mpw.dat

Run the Platform Manager Migration Utility

The Platform Manager migration utility can migrate the following:

- Password manager configuration and data.
- Users, roles, and groups.
- JAAS and port configurations.

For a general description of migration utility behavior, see ["Migration Utilities" on page 31](#).

Perform a Custom Migration

- On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory* \PlatformManager\migrate\bin directory, and run the command `migrate.{bat|}`.
- The utility asks for the full path to one of the following:
 - The Software AG directory that contains the old Platform Manager installation (for example, `C:\old_Software AG_directory` or `/opt/old_Software AG_directory`).

- The Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter N.
 4. The utility asks whether to migrate users, roles, and groups; the JAAS configuration; and password manager data.
 5. The utility asks whether to migrate HTTP and HTTPS port configurations.
 6. The utility asks whether to export your settings. If you want to perform other migrations by importing the settings from this session, enter Y. If not, enter N.
 7. The utility asks whether to begin migration. If you enter Y, the utility migrates the data you selected.

Migrate Using Imported Settings

Imported settings can come from the following:

- Settings you exported from a custom migration. These settings are stored in a file named `migrate.dat`. The file is stored in the `9.8_Software AG_directory \PlatformManager \migrate` directory from which you ran the custom migration. Copy the `migrate.dat` file to any directory on machines that host Platform Manager 9.8 installations to which you want to migrate data.
- Settings in the default migrations provided with Platform Manager. For each old release, the settings are stored in a file named `migraterelease sbs.dat` file in the `9.8_Software AG_directory \PlatformManager \migrate` directory. The settings tell the migration utility to migrate the data listed under ["Run the Platform Manager Migration Utility" on page 95](#).

Migrate Using Imported Settings with Prompting

1. On the 9.8 machine, open a command window or shell, go to the `9.8_Software AG_directory \PlatformManager \migrate` directory, and run the command `migrate.{bat|}`.
2. The utility asks for the full path to the Software AG directory that contains the old Platform Manager installation or to the Zip file you made earlier.
3. The utility asks whether to import migration settings. Enter Y and, when prompted, provide the full path to the `migrate.dat` file or the `migrate{901|951|960}sbs.dat` file.

Migrate Using Imported Settings Without Prompting

On the 9.8 machine, open a command window or shell, go to the `9.8_Software AG_directory \PlatformManager \migrate` directory, and run the command below. If an error occurs, the utility exits.

```
migrate.{bat|}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory|Zip_file}
-importFile full_path_to_{migrate.dat|migrate{960|970}sbs.dat}
-silent true
```


Complete the Platform Manager Upgrade

Restore Trusted Authentication for JAAS

If you chose to migrate your JAAS configuration, follow the steps below to restore trusted authentication.

1. Go to the *9.8_Software AG_directory* \profiles\SPM\configuration\security directory and open the /jaas.config file.
2. Add the entries below to the Default login context:

```
com.softwareag.security.idp.saml.lm.SAMLAssertValidatorLoginModule sufficient;  
com.softwareag.security.idp.saml.lm.JMXDelegatedAuthLoginModule sufficient;  
com.softwareag.security.idp.saml.lm.SAMLAssertIssuerLoginModule optional;
```

Update the Platform ManagerNERV Configuration

Go to the *9.8_Software AG_directory* \profiles\SPM\configuration \com.softwareag.platform.config.propsloader directory and open the com.softwareag.platform.eda.nerv.properties file. Update JMS provider references to point to your 9.8 JMS provider, if they do not already. Update file system paths to point to the 9.8 installation.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in ["Complete Final Upgrade Tasks for All Products"](#) on page 118.

13 Completing the API-Portal and CentraSite Upgrade

■ Completing the API-Portal Upgrade	100
■ Completing the CentraSite Upgrade	101

Completing the API-Portal Upgrade

1. If you installed API-Portal 9.8 in a clustered, high-availability setup, make sure all nodes are running and accessible to the Zookeeper ensemble. Then register all the nodes with the parent node by invoking the REST service shown below. The default for the load balancer port is 18002.

Parameter	Value
End point	<code>http://load_balancer_host:load_balancer_port/acc/rest/nodes/</code>
Method type	POST
Content type	application/json
Payload	<code>{"nodename": "host_name", "hostname": "host_name", "port": "cloud_agent_port", "username": "Cloud", "password": "g3h31m"}</code>

2. Start the 9.8 API-Portal Cloud Controller.

3. Run this command:

```
acc> startall
```

4. Restore the old master tenant into API-Portal 9.8 from the backup file you created earlier by running these commands:

```
acc> set acc config backup.restore.tenant.app.types = com.aris.umcadmin.y-umcadmin-run-prod
acc> set acc config backup.restore.tenant.app.type.names = umc
acc> restore tenant master from full_path_to_backup_file
username= your_user_name password=your_password
```

5. Earlier in this guide, you generated a list of tenants in the old API-Portal. For each old tenant on the list except the default tenant and the master tenant, create a corresponding tenant in API-Portal 9.8 by running this command:

```
cmd> cd 9.8_Software_AG_directory\API_Portal\server\bin\work\
work_apiportalbundle_m\tools\bin
cmd> y-tenantmgmt.bat -s http://load_balancer_host:load_balancer_port
-t tenant_name createTenant
-u your_user_name -p your_password
```

6. In the User Management Component at `http://host:port/umc`, import the license for each tenant you created.

7. In the 9.8 API-Portal Cloud Controller, run these commands:

```
acc> stop apiportalbundle_m
acc> start apiportalbundle_m
```

- For each new tenant you created, restore the data for the corresponding old tenant into API-Portal 9.8 from the backup file you created earlier by running the commands below. Each file has the file extension .acb.

Important: Inform API-Portal users that they cannot work on the tenants during the restoration process.

Important: If the new tenant contains any data, the data will be replaced by the data you restore from the old tenant.

```
acc> set acc config backup.restore.tenant.app.types = UMC, ABS, ADS, ECP
acc> restore tenant tenant_name from full_path_to_backup_file
username=your_user_name password=your_password
```

- Run these commands:

```
acc> stop apiportalbundle_m
acc> start apiportalbundle_m
```

- If you installed API-Portal 9.8 on the same machine as the old API-Portal, run this command:

```
acc> stopall
```

Completing the CentraSite Upgrade

Before Migrating Data

- Make sure the old CentraSite and CentraSite 9.8 are shut down.
- Export configuration data from the old CentraSite and assets from the old Registry Repository into a Zip file. On the old machine, open a command window or shell, go to the *old_Software AG_directory*/CentraSite/utilities directory, and run this command:

```
sbsExport.{cmd|sh} /full_path_to_zip_file
```

An example of this command is `./sbsExport.sh /tmp/sbs_cs82_data.zip`

- If the old and new CentraSite installations are on different machines, copy the Zip file to any directory on the 9.8 machine.

Migrate the Data

CentraSite scripts migrate configuration data from the old CentraSite and assets from the old Registry Repository to the new CentraSite installation. They also transform and migrate the LDAP configuration from the old Registry Repository to the new CentraSite JAAS configuration.

- Import configuration data and assets from the Zip file you created earlier. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory*/CentraSite/utilities directory and run this command:

```
sbsImport.{cmd|sh} /full_path_to_zip_file
```

An example of this command is `./sbsImport.sh /tmp/sbs_cs82_data.zip`

2. Start CentraSite 9.8.
3. Transform and migrate the LDAP configuration to the new JAAS configuration by running this command:

```
CentraSiteCommand.{cmd|sh} generate JaasConfiguration -url CentraSite_9.8_URL
-user CentraSite_9.8_admin_user -password password
```

The script generates JAAS LoginModule entries that correspond to the old LDAP configuration and saves the entries in the `jaas.config` file in the `9.8_Software AG_directory/CentraSite/profiles/CTP/configuration` directory. For each LDAP domain, the script creates user and group files that map internal (CentraSite) properties to external (LDAP) properties and saves the files in the `9.8_Software AG_directory/CentraSite/profiles/CTP/configuration/com.softwareag.platform.config.propsloader` directory.

4. If you use single sign-on with CentraSite, do the following:
 - a. Open the `jaas.config` file in the old and new `Software AG_directory/CentraSite/profiles/CTP/configuration` directories. Copy the following from the old file to the 9.8 file:
 - `ServletHeaderLoginModule` for extracting the user ID from the incoming HTTP header.
 - `SimpleNameMappingLoginModule`, if you are using it.
 - Any other entries you are using to process the extracted user ID.

The 9.8 `jaas.config` file should look like this:

```
CentraSite {
  com.softwareag.security.jaas.login.internal.InternalLoginModule sufficient
  ...
  com.softwareag.security.sin.is.ldap.lm.LDAPLoginModule sufficient
  ...
  com.softwareag.security.jaas.login.modules.ServletHeaderLoginModule
  required
  ...
  com.softwareag.security.jaas.login.modules SimpleNameMappingLoginModule
  required
  ...
  com.softwareag.security.sin.is.ldap.lm.LDAPLoginModule required
  ...
};
```

The initial `InternalLoginModule` is normally only for users in the INTERNAL domain, and the initial `LDAPLoginModule` is for LDAP users that are logging in directly and not via single sign-on. If you need only single-sign on logins, you can remove the initial `InternalLoginModule` and `LDAPLoginModule`.

- b. Set up your LDAP configuration to resolve the extracted user ID via LDAP. Modify the generated LDAP login module to enable single sign-on-related options, such as technical user. Apply LDAP single sign-on technical user credentials if necessary.

Complete the Upgrade

1. Change any settings in CentraSite Control that you noted in [Prepare the Old CentraSite](#) .
2. If you installed plug-ins that are GUI extensions for CentraSite Control in the old CentraSite installation, install them in the CentraSite 9.8 installation.
3. If you are using API-Portal with CentraSite, republish all API-Portal instances you created in the old release to API-Portal 9.8. For instructions, see the CentraSite documentation.
4. 8.2 upgrade: If you want to use instances of ApplinX object types from CentraSite 8.2 in CentraSite 9.8, manually re-create them using the definitions you noted in [Prepare the Old CentraSite](#) .
5. Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

14 Complete the Mobile Upgrade

■ 9.7 Upgrade: Complete the Mobile Designer Upgrade	106
■ Complete the Mobile Administrator Upgrade	106

9.7 Upgrade: Complete the Mobile Designer Upgrade

If you created apps using the Mobile Development perspective in Software AG Designer and want to continue to do so, or if you created apps using Mobile Designer but want to use the Mobile Development perspective in Software AG Designer from now on, delete the environment variable `MOBILE_DESIGNER` as instructed below.

If you created apps using Mobile Designer and want to continue to do so, and if you also want to continue to use your existing build environment (that is, an IDE other than Software AG Designer, or Jenkins), delete the `MOBILE_DESIGNER` environment variable as instructed below. Then pass the property `#Denv.MOBILE_DESIGNER=full_path_to_Mobile_Designer` to Ant when calling Mobile Designer Ant targets (for example, enter `ant #Denv.MOBILE_DESIGNER=C:\SoftwareAG\MobileDesigner +Multi-Build`).

To remove the `MOBILE_DESIGNER` environment variable, do the following:

System	Steps
Windows	Go to the Control Panel, click System , and then click Advanced System Settings . On the Advanced tab, click Environment Variables and delete the <code>MOBILE_DESIGNER</code> environment variable. Then restart your system.
Mac OS X	Delete the <code>MOBILE_DESIGNER</code> environment variable from your <code>.bash_profile</code> , your <code>/etc/launchd.conf</code> file, or both. The file or files that contain the environment variable depend on the version of MAC OS X you are using, and on modifications you might have made. Then restart your system.

Complete the Mobile Administrator Upgrade

Overview

You will complete the Mobile Administrator upgrade by:

- Running the Mobile Administrator updater to upgrade the Mobile Administrator source code and migrate the Mobile Administrator database.
- Verifying that all features required by your use case are configured.
- Updating any build nodes you used with the old Mobile Administrator.
- Updating the App Store Client.

Run the Updater and Verify Configurations

1. Go to the *Software AG_directory* /MobileAdministrator directory and run the updater by running the command `./update.sh` as root.
2. From the same directory, start Mobile Administrator by running the command `etc/init.d/appfresh-mobile start`.
3. In Mobile Administrator, click **Settings**, click **Sites**, and click a domain. On the domain page, click **Configuration Assistant**. Make sure all the features you need are configured. If necessary, follow the instructions to configure any features that are missing. Repeat this step for each domain.

Update Build Nodes

1. If you used the Mac OS build node with the old Mobile Administrator, update the build node as follows:
 - a. Go to the **Mobile Administrator > Build Nodes** page at `https://Mobile Administrator_URL/build_nodes`. Download the `appfresh-build-node-version.zip` file, and extract the files to any directory.
 - b. Launch the build node application. On the **Install** tab, in the **Build Node** field, make sure the displayed build node version corresponds to the version of the build node you just downloaded (that is, `appfresh-build-node-version.zip`).
2. If you used the Windows 8 build node with the old Mobile Administrator, update the build node as follows:
 - a. Go to the Windows Control Panel and uninstall the previous build node.
 - b. Go to the **Mobile Administrator > Build Nodes** page at `https://Mobile Administrator_URL/build_nodes`. Download and run the `AppFreshBuildNode-version.exe` file.
 - c. Launch the build node application. On the **Build Node** tab, start the build node by clicking **Start**.

Update the App Store Client

1. Go to the `https://Mobile Administrator_URL/apps/appfresh-mobile-client/build` page.
2. On the **Build Configurations** tab, start the build jobs for the App Store Client by selecting the platforms for which to update the client and clicking **Launch Selected Build Configs**.
3. On the **Build Jobs** tab, verify that the build jobs have completed successfully. Corresponding build nodes should appear on the `https://Mobile Administrator_URL/build_nodes` page. If they do not, click on the build job to display the details that will help you diagnose the problem.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

15 Complete the OneData Upgrade

■ Run the OneData Migration Utility	110
■ Migrate Customizations to the Java Service Wrapper	110
■ 8.4 Upgrade: Enable New Logging Method	110
■ Complete Final Upgrade Tasks	110

Run the OneData Migration Utility

The OneData migration utility migrates configuration data.

1. Make sure the old OneData is shut down.
2. On the 9.8 machine, open a command window or shell, go to the *9.8_Software AG_directory \profiles\ODE\bin\migrate* directory, and run the command below. If an error occurs, the utility exits.

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory|Zip_file}
-silent true
```

Before migrating data, the utility backs up the files that contain the 9.8 configuration data in the *9.8_Software AG_directory \profiles\ODE\bin\onedata\config* directory. The backup files have the extension `.bck`.

Migrate Customizations to the Java Service Wrapper

If you customized the old OneData Java Service Wrapper, copy the customizations from the old `wrapper_conf` or `custom_wrapper.conf` files to the `9.8 custom_wrapper.conf` file. The files are located in the *Software AG_directory \profiles\ODE\configuration* directory.

8.4 Upgrade: Enable New Logging Method

In OneData 8.4, log entries for an action might have been distributed across multiple files. Starting in OneData 9.0, to improve debugging, log entries for an action are written to a single file. You must enable the new method of logging for OneData modules. Follow the examples in the `sample_log4j_config.txt` file in the *9.8_Software AG_directory \profiles\ODE\bin\onedata\config* directory.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

16 Completing the Presto Upgrade

■ Overview	112
■ Set Up the Presto Repository	112
■ Update the Presto Configuration and Copy Extensions	113
■ 3.7 or 3.8 Upgrade: Upgrade the Integrated MashZone Server and RTBS Server Configurations, and MashZone Feeds	114
■ Execute Upgrade Commands	114
■ 3.6 or 3.7 Upgrade: Update Mashups that Use RAQL Queries	115
■ Update the Snapshots Repository Connection Information	115
■ Complete Final Upgrade Tasks	116

Overview

This chapter explains how to complete an upgrade from Presto 3.6, 3.7, or 3.8 to Presto 3.9. If you want to upgrade from releases of Presto prior to 3.6, contact Software AG Global Consulting Services.

Presto stores all of its data, including mashups, mashables, apps and workspaces, in the Presto repository. To upgrade, you will set up Presto 3.9 to use the old Presto repository. If you deployed extensions to features of the old Presto (for example, if you used scripting in mashups and the scripts are stored in files), you will also copy the extensions to the Presto 3.9 installation.

Set Up the Presto Repository

Set Up Repository When Using the Embedded Derby Database

If you used the embedded Derby database for your old Presto repository, follow the instructions below.

1. Presto 3.9 comes with a prepopulated Derby Presto repository in the `prestorepository` directory. Go to the `9.8_Software AG_directory /Presto/apache-tomee-jaxrs/bin` and rename the `prestorepository` directory (for example, rename the directory `prestorepository_orig`).
2. 3.7 or 3.8 upgrade: Go to the `old_Software AG_directory /Presto/apache-tomee-jaxrs/bin` directory and copy the `prestorepository` and `mashzonerepository` directories to the `9.8_Software AG_directory /Presto/apache-tomee-jaxrs/bin` directory.
3. 3.6 upgrade: Go to the `Presto3.6.1/Presto-3.6.1-20130930/prestorepository/derby` directory and copy the `prestorepository` directory to the `9.8_Software AG_directory /Presto/apache-tomee-jaxrs/bin` directory.

Set Up Repository When Using Any Other Database

If you used MySQL, Oracle, PostGRES, or SQL Server for your old Presto repository, follow the instructions below.

1. Go to the `old_Software AG_directory /Presto/apache-tomee-jaxrs/lib` directory and copy the JDBC driver jar file for your database to the same directory in the Presto 3.9 installation.

Database	JDBC Driver Jar File
MySQL	mysql-connector-java-5.1.31.jar

Database	JDBC Driver Jar File
Oracle	ojdbc6.jar
PostGres	postgresql-9.2.1004.jdbc4.jar
SQL Server	jtds-1.3.1.jar

2. If you used MySQL, Oracle, or SQL Server for your old Presto, do the following:
 - a. Go to the *9.8_Software AG_directory /Presto/apache-tomee-jaxrs/webapps/presto/WEB-INF/lib* directory and delete the `jackbe-presto-rds-postgre-derby-3.9.jar` file.
 - b. Go to the *9.8_Software AG_directory /Presto/prestorepository* directory and copy the `jackbe-presto-rds-oracle-mysql-mssql-3.9.jar` file to the *9.8_Software AG_directory /Presto/apache-tomee-jaxrs/webapps/presto/WEB-INF/lib* directory.
3. 3.7 or 3.8 upgrade: Go to the *old_Software AG_directory /Presto/apache-tomee/jaxrs/conf* directory and copy the `tomee.xml` file to the same directory in the Presto 3.9 installation.
4. 3.6 upgrade: Configure Presto 3.9 to use the existing repository. For instructions, go to the Presto 3.9 Presto Administration documentation, and then to Getting Started with Presto, and then to Move the Presto and Mashzone Repositories to a Robust Database Solution. Skip steps 1 through 4, and start with step 5.

Update the Presto Configuration and Copy Extensions

Most of the Presto configuration is stored in the Presto repository. This section explains how to copy the parts of the old configuration that were not maintained in the repository, including extensions such as custom security profiles that you deployed in the old Presto. The procedure varies depending on whether you used an external configuration directory with the old Presto.

Update When an External Configuration Directory Was Used

1. Update the classpath in the application server that hosts Presto 3.9 to work with the external configuration directory and any subdirectories. For instructions, see the topic on setting up an external configuration directory in the Presto Administration documentation.
2. Manually migrate the changes you made in the old configuration files listed below from the old Presto to the corresponding files in Presto 3.9. Because of the complexity of the configuration, the easiest method is to use the difference tool offered by many text editors.
 - All files listed in the topic on configuration files that must be internal in the Presto Administration documentation.

- The `userRepositoryLdap.properties` file, located in the external configuration directory.

Update When No External Configuration Directory Was Used

1. Copy any extensions you deployed from the old Presto to the corresponding directory for Presto 3.9. See the Presto Administration documentation for a complete list of extensions you might need to copy.
2. Manually migrate the changes you made in the old configuration files listed below from the old Presto to the corresponding files in Presto 3.9. Because of the complexity of the configuration, the easiest method is to use the difference tool offered by many text editors.
 - All files listed in the section on configuration files that must be internal in the Presto Administration documentation.
 - The `userRepositoryLdap.properties` file, located in the `old_Software AG_directory / Presto/webapps/presto/WEB-INF/classes` directory.
3. If you used database mashables, go to the `old_Software AG_directory / Presto/apache-tomee-jaxrs/presto/WEB-INF/config/db/jdbc` directory. Copy the database driver jar files to the same directory in the Presto 3.9 installation.

3.7 or 3.8 Upgrade: Upgrade the Integrated MashZone Server and RTBS Server Configurations, and MashZone Feeds

Presto's Integrated MashZone Server stores additional data. Follow the instructions below to copy the data to the Presto 3.9 installation.

1. Go to the `old_Software AG_directory / Presto/mashzone/data/resources` directory and copy the files to the same directory in the Presto 3.9 installation.
2. Go to the `old_Software AG_directory / Presto/mashzone/data/jdbcdrivers` directory and copy the files to the same directory in the Presto 3.9 installation.
3. Go to the Admin Console in Presto Hub. Expand the MashZone section, click **Server Settings**, and click the **Resources** tab. Review the paths to resource directories. If any paths point to directories in the old Presto installation, copy those directories to the Presto 3.9 installation and then update the paths in MashZone Administration to point to the correct location in the Presto 3.8 installation.

Execute Upgrade Commands

Start the Presto server. Open a command window or shell, go to the `9.8_Software AG_directory \Presto\prestocli\bin` directory, and run these commands:

```
.\padmin.bat runRequestFile -u Administrator -c -w manage
```

```
-f ..\..\upgrade\3.9\upgrade_3.9.0.jump.txt
.\padmin.bat importPolicy -u Administrator -w manage
-f ..\..\upgrade\3.9\Presto_RO_Policy.xml
```

3.6 or 3.7 Upgrade: Update Mashups that Use RAQL Queries

Presto 3.8 and later contain significant changes and additions to the RAQL query syntax that you use in mashups. If you have mashups from Presto 3.7 or earlier, you must update the mashups that use RAQL queries to run on Presto 3.9, but you might want to also update the other mashups to take advantage of new RAQL query syntax. For complete information, see the topic about migrating old RAQL queries to 3.9 in the Presto documentation.

Update the Snapshots Repository Connection Information

If you used Snapshots in your old Presto and you stored the data in a separate Snapshots repository, follow the steps below.

1. Open the Presto 3.9 Admin Console.
2. If the JDBC driver you used with the old Snapshots repository is not the same driver you used for the old Presto repository, go to the *old_Software AG_directory/Presto/apache-tomee-jaxrs/lib* directory. Copy the JDBC driver jar file for the database that contains the Snapshots repository over the same file in the same directory in the Presto 3.9 installation.

Database	JDBC Driver Jar File
Derby	derby-embedded-10.5.3.jar
MySQL	mysql-connector-java-5.1.31.jar
Oracle	ojdbc6.jar
PostGres	postgresql-9.2.1004.jdbc4.jar
SQL Server	jtds-1.3.1.jar

3. Update the Snapshots 3.9 configuration to point to the Snapshots 3.9 repository as follows:
 - a. Open the **JDBC Configuration** menu, click **Datasources**, click **SnapshotDatasource**, and then click **Edit**.

- b. Edit the configuration properties as needed. For information on the properties, see the section on adding a data source in the Presto Administration documentation.
- c. Click **Save**.

Complete Final Upgrade Tasks

Complete final upgrade tasks as instructed in "[Complete Final Upgrade Tasks for All Products](#)" on page 118.

17 Completing Final Upgrade Tasks for All Products, and Troubleshooting

■ Complete Final Upgrade Tasks for All Products	118
■ Troubleshooting	119

Complete Final Upgrade Tasks for All Products

- Configure your products as described in the 9.8 product documentation.
- If you installed your 9.8 products on a different machine than your old products, make sure to update host names in your 9.8 products, in the connections between your products, and in your database tables. This guide indicates many locations in which to update host names, but make sure you also specify the correct host names when you configure your 9.8 products using the instructions in the product documentation. Also make sure any absolute paths in the 9.8 configuration files point to valid locations, or change them to be correct for the new machine. If the machine has a different operating system or hardware, make sure your JVM settings are correct.
- For critical information, known issues, fixes, resolved issues not rolled into fixes, and usage notes for your products, see the product readmes for each release from your old release up to and including the 9.8 release. Readmes are available on the Software AG Documentation website.
- For changes to product behavior, services, parameters, properties, APIs, documentation, and terminology, see the product readmes. You might need to modify product files or assets to accommodate these changes. In the 9.8 readmes, the sections that list changes are cumulative and contain information for releases 9.6, 9.7, and 9.8. Read the product readmes for each of the releases from your old release up to and including the 9.8 release. For example, if you are upgrading from release 8.2, read the product readmes for releases 8.2, 9.0, 9.5, and 9.8. Readmes are available on the Software AG Documentation website.

Note: Readmes list items that are deprecated, such as services, parameters, properties, and APIs. Deprecated items are still supported by Software AG, but they will likely be removed from the product at some point in the future. Software AG recommends that you avoid using deprecated items in new functionality and solutions you develop.

- For information on new functionality, read the product release notes for each release from your old release up to and including the 9.8 release. Release notes are available on the Software AG Documentation website.
- After installation, you might have set Windows services for products to Manual, and disabled scripts that start UNIX daemons, to avoid automatically starting both old and new products. When your new environment is ready, after you stop running the old products and when you want to start running the new ones, you can reset the Windows services to Automatic, and re-enable the UNIX scripts.

Troubleshooting

If you encounter errors during the upgrade, try the troubleshooting methods below.

- Look in log files.

Type of Log	Location
Installation and uninstallation	<i>9.8_Software AG_directory</i> \install\logs directory
Product	In a logs directory in the product file structure, or in the <i>9.8_Software AG_directory</i> \profiles\product\logs directory.
Data migration	<i>9.8_Software AG_directory</i> \install\logs directory and product directories indicated by the upgrade procedures
Database migration	<i>9.8_Software AG_directory</i> \common\db\logs directory.

- Go to the [Knowledge Center](#) on the Empower Product Support Website and search using keywords such as upgrade.
- Look up error messages in the *webMethods Error Message Reference*.
- Go to the [Software AG Developer Community](#) and join the upgrade discussion forum.

If you cannot resolve the problem using the methods above, contact Software AG Global Support.