



# Enterprise Installation Guide

JDA® Enterprise Store Operations

Release 2018.1.0.0



## Legal notice

### **Rights to the content of this document**

Copyright © 2003-2018 JDA Software Group, Inc. All rights reserved.

Printed in the United States of America.

Reproduction of this document or any portion of it, in any form, without the express written consent of JDA Software Group, Inc. ("JDA") is prohibited.

These materials are protected by the Copyright Act of 1976, as amended, as an unpublished work and the foregoing notice and legend shall not be deemed to constitute publication or an intent to publish thereunder. These materials are proprietary and confidential information of JDA and may be disclosed and used only as authorized in a signed, written agreement controlling such disclosure or use.

The fact that a particular name or logo does not appear on this notice does not constitute a waiver of any intellectual property rights that JDA has established in any of its products, feature or service names, or logos.

### **Trademark, Registered, and Service Mark notices**

JDA is a registered trademark of JDA Software Group, Inc. JDALearn is a service mark of JDA Software Group, Inc.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. SAP and SAP HANA are trademarks or registered trademarks of SAP SE in Germany and in several other countries. Microsoft, Encarta, MSN, and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Autodesk and Revit are registered trademarks or trademarks of Autodesk, Inc., in the USA and other countries. All other product names and company names may be the trademarks/service marks or registered trademarks/service marks of their respective owners.

Throughout this document, certain designations may be used that are trademarks that identify the goods of third parties. Although this document attempts to identify the particular trademark owner of each mark used, the absence of a trademark symbol or other notations should not be taken as an indication that any such mark is not registered or proprietary to a third party. Use of such third-party trademarks is solely for the purpose of accurately identifying the goods of such third party. The information contained herein is subject to change without notice.

### **Modifications to the contents of this document**

JDA reserves the right, at any time and without notice, to change these materials or any of the functions, features, and specifications of any of the software described herein. JDA shall have no warranty obligation with respect to these materials of the software described herein, except as provided in the JDA software license agreement with an authorized licensee.

### **Rights to the functionality of this document**

Described functionality may not be available as part of a customer's maintenance agreement or the JDA Investment Protection Program. New features and products are subject to license fees. JDA warranty and support obligations apply only to the documentation as delivered by JDA, and are void if the documentation is modified or supplemented by anyone other than JDA. This document embodies JDA valuable trade secrets, is confidential to JDA, and must be kept in confidence and returned upon the expiration or termination of your JDA license agreement. You are not permitted to copy, extract, distribute, transfer, or share the contents of this document with anyone except authorized individuals within your organization.

### **Technical documentation**

NOTICE: This design or technical documentation is supplied as a courtesy only and does not form part of the "Documentation" as defined in your JDA license agreement. This design or technical documentation is supplied in the English language only and is supplied "as is" and without warranties. JDA, at its discretion, may choose to offer this document in additional languages, but is under no obligation to do so. JDA undertakes no obligation to update this design or technical documentation.

### **Patents**

This product may be protected by one or more United States and foreign patents. Please see the JDA Patents [website](#).

## Provide feedback on this document

JDA values your opinion and strives to ensure that the documentation you receive is clear, concise, and provides the appropriate information required for you to use each JDA application efficiently.

If you would like to provide feedback on this document, you can submit your questions or suggestions to the [JDA Documentation Management team](#) and they will be forwarded to the appropriate development teams for review and consideration in a future release.

In addition to the provided documentation, many additional resources are available to help you understand and work with your JDA applications. For more information on these resources, see the [JDA Services website](#).

## Software implementation advisement

JDA products offer intuitive solutions to solve complex problems and situations. Your software comes with highly advanced algorithms that use techniques that may include heuristics, optimization, machine learning, data science, and so on. JDA also invests heavily to ensure that the algorithms, workflows, and overall user experience can be configured to satisfy your needs, providing personalized optimization combined with extreme scalability. With the broad range of functional and technical capabilities, along with numerous configuration possibilities that the packaged solutions offer, it is commonly advised to use strategic services to design, technical services to install and tune, and education services to train, when rolling out any advanced software.

JDA recommends using JDA certified consultants to get the best possible domain and functional advice available to ensure that your products are configured to provide the best possible results. In the absence of expertise, an incorrectly configured system can lead to suboptimal results and subpar performance. In these circumstances, JDA Support Services will refer you to consulting partners who can be engaged to answer further questions and guide the implementation.

## Software support

The JDA Solution Investment Policy includes the levels of support available for your licensed applications to maximize your benefit. Through the JDA Customer Support [website](#), you can access a comprehensive summary of your licensed applications and the Solution Investment Policy that describes your current levels of support. By supporting only the newest software versions, JDA can provide you with exemplary service, enabling you to realize an evolving return on your software investment.

# Table of Contents

<b>Chapter 1. SQL Server</b> .....	<b>1</b>
Installation summary .....	1
MS clustering .....	1
SQL named instances .....	1
Prerequisites .....	1
SQL collation .....	1
SQL object names .....	1
SQL authentication .....	2
Post installation requirements .....	3
<b>Chapter 2. Retail initial database deployment</b> .....	<b>9</b>
Summary .....	9
Prerequisites .....	9
Install files .....	9
Configure SQL logins .....	10
Create the ORB database .....	10
Inject Retail code into the OR database .....	11
Create Retail databases using the Retail empty databases .....	11
Change database read-only property to False .....	12
Generic deploy .....	12
Grant retail database user access .....	14
<b>Chapter 3. Internet Information Server (IIS) configuration</b> .....	<b>15</b>
Summary .....	15
Prerequisites .....	15
Configure IIS .....	15
<b>Chapter 4. Web or application deployment</b> .....	<b>20</b>
Prerequisites .....	20
Files needed .....	20
Windows Server 2012 .....	20
Database connection authentication .....	20
Environment naming .....	20
Object repositories .....	21
Enterprise Deploy Tool notes .....	21
Web or application server initial deployment .....	21
Web site creation .....	26
Application Deploy – Host Management Server .....	26
Local Deploy – Enterprise Deploy Tool .....	29
Application testing .....	31
APE Controller support .....	32
<b>Chapter 5. Hosting Management server</b> .....	<b>34</b>
Hosting Management Server .....	34
Configuration Requirements .....	34
Prerequisites .....	34
DB Server .....	34
Retail software files needed .....	35
Windows Server 2012 .....	35
Database connection authentication .....	35
No access to <a href="http://support.bluecube.com">http://support.bluecube.com</a> .....	35
Database configuration .....	35
Web server configuration .....	37

Web site creation .....	38
Initial HM Deployment .....	39
IIS configuration .....	40
Database update .....	41
Troubleshooting HM install .....	44
Globalsettings.xml for HM server .....	44
ORBViews.sql .....	45
<b>Chapter 6. Hosting management upgrade .....</b>	<b>48</b>
Conversionreadme.txt .....	48
Hosting Management module configuration .....	48
Restore the hosting management empty database .....	48
Application code injection .....	49
Database conversion .....	49
Update ORB views .....	50
Web deploy .....	50
Web site settings .....	51
<b>Chapter 7. Upgrade: Database Conversion .....</b>	<b>52</b>
Summary .....	52
Prerequisites .....	52
Downgrading .....	52
Convert database .....	52
Upgrade files .....	52
Update the OR database .....	53
Restore the empty databases .....	53
Shutdown the application .....	54
Take backup of databases to be upgraded .....	55
Start database conversion .....	55
<b>Chapter 8. Upgrade: web or application server conversion .....</b>	<b>57</b>
Hosting management server .....	57
Remote Deployment tool .....	57
Code deployment using HMS .....	57
Deployment testing .....	58
<b>Chapter 9. Retail (classic) web client requirements .....</b>	<b>60</b>
Retail (classic) required internet Explorer settings .....	60
<b>Chapter 10. Login Security Configuration .....</b>	<b>62</b>
Recommended Security settings .....	62
<b>Appendix A. APE queue segmentation .....</b>	<b>63</b>
Benefits .....	63
Segmentation types .....	63
Primary types .....	63
Isolated segmentation examples .....	64
Segmentation by task .....	64
Segmentation by thread count .....	64
Crossover segmentation example .....	65
Segmentation by priority .....	65
Identify jobs requiring segmentation .....	66
Determine segmentation model .....	66
Identify monikers and wild card filters .....	66
Determine desired rules and thread counts .....	67
Add queue segments and filters .....	67

Add APE rules .....	68
Validate segmentation .....	68
<b>Appendix B. Feature configuration .....</b>	<b>70</b>



# Chapter 1. SQL Server

## Installation summary

This chapter of the installation guide presents deployment steps of Microsoft SQL Server 2016 or 2014 on a Windows Server 2012R2/2016 server.

## MS clustering

MS SQL Cluster Installations are not addressed in this document. At a minimum, after the application databases have been deployed, fail over the cluster and run the generic deploy package again to install required components on the secondary node.

## SQL named instances

If SQL is installed as a Named Instance, you will need to create a 32bit SQL alias for the Retail ScriptRunner program that manages database conversions.

## Prerequisites

Prerequisites provide the minimum requirements for the application. The release notes for a specific version of the application may contain additional requirements. Microsoft SQL Server requirements must be met as well. The purpose of this document is to provide core installation procedures only.

- Windows Server 2012R2 or Windows Server 2016(Standard or Data Center edition) 64 bit.
- If using a Domain User Account for the SQL Service account, it is recommended to log into the server and run the installation with this account instead of your own user account.
- The Disk drive(s) for the production application databases should be formatted with a 64K block size. For that reason, the SQL installation files and data files should not be on the same drive.

## SQL collation

The Retail releases prior to 2009.2.0 required the use of an alternate collation setting (**SQL\_AltDiction\_Pref\_CP850\_CI\_AS**) as described within this document. Starting with the 2009.2.0 releases, the SQL default collation (**SQL\_Latin1\_General\_CP1\_CI\_AS**) is supported and should be used for all new implementations. Clients with existing Enterprise databases built with the alternate collation should continue to use the alternate collation. When converting databases that include an empty set of databases, a collation check is performed. For servers on the alternate collation, the empty databases will be converted to the alternate collation. On servers configured with the default collation, no action is taken.

## SQL object names

JDA requires the use of Regular Identifiers for all object names including servers, databases, tables, stored procs, triggers, views, etc.

The rules for the format of regular identifiers depend on the database compatibility level. When the compatibility level is 90 or higher, the following rules apply:

The first character must be one of the following:

- A letter as defined by the Unicode Standard 3.2. The Unicode definition of letters includes Latin characters from a through z, from A through Z, and also letter characters from other languages.
- The underscore (\_), at sign (@), or number sign (#).

- Certain symbols at the beginning of an identifier have special meaning in SQL Server. A regular identifier that starts with the @ sign always denotes a local variable or parameter and cannot be used as the name of any other type of object. An identifier that starts with a number sign denotes a temporary table or procedure. An identifier that starts with double number signs (##) denotes a global temporary object. Although the number sign or double number sign characters can be used to begin the names of other types of objects, we do not recommend this practice.
- Some Transact-SQL functions have names that start with double @ signs (@@). To avoid confusion with these functions, you should not use names that start with @@.

Subsequent characters can include the following:

- Letters as defined in the Unicode Standard 3.2.
- Decimal numbers from either Basic Latin or other national scripts.
- The at sign (@), dollar sign (\$), number sign(#), or underscore(\_).
- The identifier must not be a Transact-SQL reserved word. SQL Server reserves both the uppercase and lowercase versions of reserved words.
- Embedded spaces or special characters are not allowed.
- Supplementary characters are not allowed.
- When identifiers are used in Transact-SQL statements, the identifiers that do not comply with these rules must be delimited by double quotation marks or brackets.

## SQL authentication

JDA Retail applications currently require mixed mode authentication. The application itself can use SQL user accounts or SQL integrated accounts, but supporting the application requires additional SQL user accounts.

- The SQL Server and Agent services accounts must be granted explicit access to SQL (group membership is not sufficient). This account must also have Modify access to the root directory of the drive where the archive database data files will be stored.
- A sysadmin privileged SQL user account must be created to support the deploy process. This account must remain active at all times.
- CRP, an optional feature in the application, requires an SQL user account.
- An SQL user with db\_datareader to all application databases is required for use by the Host Management application and by support personnel.
- Typical names: there is no specific naming requirement. The names used in prod and non-prod can be either the same (use different passwords) or different. Using the same names is cleaner and easier to support:
  - use: sysadmin for ScriptRunner program. Only needed for the deploy process, may otherwise be disabled or deleted till the next deploy.
  - retail\_app\_user: application user account when using SQL Authentication
  - retail\_crp\_user: CRP feature database access account
  - retail\_ro\_user: Read Only user account for HM and Support

## Post installation requirements

### Database mail

This document provides instructions to configure SQL Database Mail and SQL Agent Alerts for sending SQL alerts. You will need to know the SMTP server address and security requirements to configure this feature.

### Microsoft SQL Server 2005 backward compatibility components

This document contains the instructions to install this component. There are 32 bit and 64-bit editions and the only version information available is the Month and Year displayed the top of the Microsoft Download page. Use **msi titled: SQLServer2005\_BC\_x64.msi**

([http://download.microsoft.com/download/2/7/c/27c60d49-6dbe-423e-9a9e-1c873f269484/SQLServer2005\\_BC\\_x64.msi](http://download.microsoft.com/download/2/7/c/27c60d49-6dbe-423e-9a9e-1c873f269484/SQLServer2005_BC_x64.msi))

However, in some systems, we still see an error when trying to launch ScriptRunner. If the system reports an error for an ActiveX control called MSCOMCTL.OCX, you will need to register the file manually. If you cannot find the file, contact a JDA Implementation Engineer for a copy. For SQL Clusters, repeat this on each DB node in the cluster.

### *SQL server installation*

1. Click **SQL Server setup**.
2. Click **Installation** and **New SQL Server**. The **SQL Server** setup checks for pre-requisites. These may already be loaded by standard update processes.
3. Click **OK**. A confirmation window is displayed asking you to install a hotfix.
4. Click **OK**. A Windows Update window is displayed.
5. Click **Restart** now to reboot the server and complete the updates.
6. When the reboot completes, launch the setup again.
7. Perform one of the following:
  - If all operations completed and passed, click **OK**. The Product Key window is displayed.
  - Verify any failed or warning operations before continuing.
8. In the **Product Key** field, enter your product key and click **Next**. The License Terms window is displayed.
9. Select the **I accept the license terms** check box and click **Next**. The Setup Support Files window is displayed.
10. Click **Install**. The Setup Support Rules window is displayed.
11. Perform one of the following:
  - If there are warnings or a failure, verify the messages and click **Re-Run**.

- To continue the installation, click **Next**. The Feature Selection window is displayed.

**Note:** If the Windows Firewall Service is used, be sure it has been configured properly to support SQL. In particular are the SQL Server ports for data access and database discovery as well as the MS Distributed Transaction Coordinator.

12. Select the following:

- **SQL Server Database Services:** Required
- **SQL Server Replication**
- **Reporting Services:** Optional
- **Client Tools Connectivity**
- **Client Tools Backwards Compatibility**
- **Management Tools** – Basic
- **Management Tools** – Complete
- **Shared feature directory:** modify as needed

13. Click **Next**. The Instance Configuration window is displayed.

**Note:** JDA recommends changing the installation path for all components to a non-system drive. It is not recommended to install the Analysis server. The RPM application that uses the Analysis Server must be installed on a separate SQL server.

14. Change the **Instance root directory** to a non-system drive. A 64K block size for this drive is recommended.

**Note:** JDA can use an SQL installation using a **Named instance**; an SQLAlias is required to support the database conversion process. Please review the installation notes at the beginning of this document on using Named Instances.

15. Click **Next**. The Disk Space Requirements window is displayed.

16. Verify disk usage and click **Next**. The Server Configuration window is displayed.

17. Enter the account credentials, change the **SQL Server Agent Startup Type** to **Automatic** and click the **Collation** tab.

18. New installations should use the *default collation*. If you are reinstalling for an existing environment that used the alternate collation, you must select **Database Engine > Click Customize > SQL\_AltDiction\_Pref\_CP850\_CI\_AS**.

19. Click **Next**. The Database Engine Configuration window is displayed.

20. Select the following options on the Account Provisioning tab:

- **Authentication Mode:** Mixed Mode
- **Built-in SQL Server system administrator account password:** enter the system administrator account password

- **Specify SQL Server administrators** > enter at least one user account or group
21. Click the **Data Directories** tab.
  22. Enter the required information and click **Next**. The Reporting Services Configuration window is displayed.
 

**Note:** JDA recommends separate drives for the following directories:

    - User database
    - User database log and Temp DB log
    - Temp DB
    - Backup
  23. Select the **Install the native mode default configuration** option and click **Next**. The Error and Usage Reporting window is displayed.
  24. Click **Next**. The Installation Rules window is displayed.
  25. Perform one of the following:
    - If all operations completed and passed, click **Next**. The Ready to Install window is displayed.
    - Verify any failed or warning operations before continuing.
  26. Click **Install**. The Installation Progress window is displayed.
  27. Click **Next**. The Complete window is displayed.
  28. Click **Close**.
  29. Start SQL Server Management Studio by selecting **Start > All Programs > Microsoft SQL Server <Version Number> > SQL Server Management Studio**.
  30. Right-click **SQL Server** and select **Facets**. The View Facets window is displayed.
  31. In the **Facet** field, select **Surface Area Configuration**.
  32. Set the following properties to true:
    - **Ad Hoc Remote Queries**
    - **CLR Integration**
    - **Database Mail**
    - **SQL Mail**

**Note:** The **xp\_cmdshell** and **OLE Automation** properties are no longer required to be set to true as part of a security-related improvement.
  33. Click **OK**.
  34. Right-click **SQL Server** and select **Facets**. The View Facets window is displayed.
  35. In the **Facet** field, select **Server Configuration**.

36. Set the **MaxDegreeofParallelism** property. If the system has 4 or more CPU cores, set to 4. Otherwise set to number of available CPU cores.
37. Click **OK**.
38. Start SQL Server Management Studio by selecting **Start > All Programs > Microsoft SQL Server <version\_number> > SQL Server Management Studio**.
39. Connect to the local server, right-click **Replication** and select **Configure Distribution**. The Configure Distribution Wizard is displayed.
40. Click **Next**. The Distributor window is displayed.
41. Select **[Server Name] will act as its own Distributor; SQL Server will create a distribution database and log** and click **Next**. The SQL Server Agent Start window is displayed.
42. Select **Yes** and click **Next**. The Snapshot Folder window is displayed.
43. This path should be left alone and click **Next**. The Distribution Database window is displayed.
44. Do not change the database name, but you may change the database path and log path and click **Next**. The Publishers window is displayed.
45. Click **Next**. The Wizard Actions window is displayed.
46. Select **Configure distribution** and click **Next**. The Complete the Wizard window is displayed.
47. Click **Finish**. The Configuring... window is displayed.
48. When all items are complete, click **Close**.

**Note:** If the SQL Service account used does not have Administrator access rights, it will not be able to set the SQL Server Agent service properties. In this case, adjust the service properties manually.

If the database mail is not being used, skip to step 66.

49. Start SQL Server Management Studio by selecting **Start > All Programs > Microsoft SQL Server <Version Number> > SQL Server Management Studio**.
50. Expand **Management**, right-click **Database Mail** and select **Configure Database Mail**. The Welcome to Database Mail Configuration Wizard window is displayed.
51. Click **Next**. The Select Configuration task window is displayed.
52. Select the **Set up Database Mail** option and click **Next**. The New Profile window is displayed.
53. In the **Profile Name** field, enter the SQL Server Service account name and click **Add**. The New Database Mail Account window is displayed.
54. Enter the following information:
  - **Account name:** use the SQL Server services account
  - **Description:** as desired
  - **E-mail address:** this is the desired sender address, be sure this is a verified address.
  - **Display Name:** as desired

- **Reply e-mail:** as desired
  - **Server name:** enter the name of the local SMTP server.
55. Click **OK**. The **New Profile** window is displayed.
  56. Click **Next**. The Manage Profile Security window is displayed.
  57. Select **Public**, set the **Default Profile** to Yes and click **Next**. The Configure System Parameters window is displayed.
  58. In the **Maximum File Size (Bytes)** field, enter 9000000 and click **Next**. The Complete the Wizard window is displayed.
  59. Click **Finish**. The Configuring window is displayed.
  60. Click **Close**.
  61. In SQL Server Management Studio, right-click **Database Mail** and select **Send Test E-Mail**.
  62. Right-click **SQL Server Agent** and select **Properties**. The SQL Server Agent Properties window is displayed.
  63. Enter the following information:
    - Select **Alert System**
    - **Enable mail profile**
    - **Mail Profile** select the profile
    - Select **Replace tokens for all job responses to alerts**
  64. Click **OK**.
  65. Right-click **SQL Server Agent** and select **Restart**.
  66. Run **SQLSERVER2005\_BC.msi**. A message is displayed asking if you want to run the file.
  67. Click **Run**. The Welcome to the Install Wizard for Microsoft SQL Server 2005 Backward compatibility window is displayed.
  68. Click **Next**. The License Agreement window is displayed.
  69. Select **Accept** and click **Next**. The Registration window is displayed.
  70. Enter the details and click **Next**. The Feature Selection window is displayed.
  71. Enable all features and click **Next**. The Ready to Install the Program window is displayed.
  72. Click **Install**. The Complete the Microsoft SQL Server 2005 Backward compatibility Setup window is displayed.
  73. Click **Finish**.
  74. Configure DTC Access:
    - a. Navigate to **Administrative Tools>Component Services**.

- b. Expand **Component Services>Computers>My Computer>Distributed Transactions**.
  - c. Right-click **Local DTC** and select **Properties>Security**.
  - d. Enable the following:
    - Network DTC Access
    - Allow Remote Administration
    - Allow Inbound
    - Allow Outbound
    - No Authentication Required
    - Enable XA Transactions
  - e. Click **OK**. The DTC Console Message is displayed.
  - f. Click **Yes**. The MS DTC restarts.
  - g. Click **OK**.
75. Close **Component Services**.
76. If the server is a member of a cluster:
- Move the MSDTC resource to the passive node of the cluster.
  - Use the above instructions to verify the MSDTC Security settings are the same. If not, correct them.
  - Move the MSDTC resource back to the original node.
  - Take the SQL Server resource(s) off-line and bring it back on-line.



# Chapter 2. Retail initial database deployment

## Summary

This chapter of the installation guide presents the initial setup of the Retail databases. A Retail environment consists of a set of databases as follows:

DB Type	DB Name (oltp, orb can be any name)
Object Repository database	orb
OLTP (Online Transaction Processing)	retail
Archive	retail_archive
Archive Monthly Template	retail_archive_yyyy_mm
ODS (Operational Data Store)	retail_wh
Point Of Sale	retail_pos
REFS (Retail External Facing Services)	Retail_refs

## Prerequisites

- MS SQL Server 2016 SP1, 2014 SP2 (Standard or Enterprise Edition), 64-bit
- **Environment name:** select an environment that will be used for the application. This name should be consistent with the following uses to minimize confusion in multiple environment locations. One exception is the use of the hyphen and underscore. Database names cannot use the hyphen and the underscore, used in a URL creates confusion when applications automatically underline URLs. Because of this, the underscore is typically used in the database names, but a hyphen is used in all other names:
  - URL: **retail.domain.com**
  - Website name: **retail**
  - Name Space: **retail**
  - Database names: **retail, retail\_archive, retail\_archive\_yyyy\_mm, retail\_pos, retail\_wh**

## Install files

You will need the following to complete a full installation:

Generic Deploy package, the Application Code (.EXE file) and the Database Conversion (.zip) and the Retail Installer (.MSI) for Persona functions.

- **ORB:** empty Object Repository database
1. Obtain the desired Retail software version from your JDA representative. Internally this is obtained from the Software Replication Lab.
  2. Move all of the downloaded files into a consolidated Retail Database folder.
  3. Extract each of the zip files to individual folders.

4. Create a folder for each build you deploy as well as for any updated Generic Deploy and/or Generic Replication packages.

## Configure SQL logins

SQL login IDs must be configured to manage the Retail databases and to grant access by the application.

1. Open **SQL Server Management Studio Security**.
2. Right-click **Logins** and select **New Login**. The Login - New window is displayed.
3. Enter a login name. For example, RETAILDBO or Deploy.
4. Select **SQL Server Authentication**.
5. Select the appropriate Password Policies.
6. Click **Server Roles**, select **sysadmin** and Click **OK** to grant the user sysadmin rights.
7. Create a second account, but do not grant any rights at this time. Once the Enterprise databases have been created, this account will be given db owner rights to the databases.

## Create the ORB database

The Object Repository database is used by the WEB and APP servers to access application code.

The .EXE file in the build folder contains the new object code to be injected into the OR Database.

Extract the database BAK file from orb\_{appropriate version} compressed file location to restore the Object Repository.

1. Open **SQL Server Management Server**.
2. Right-click **Databases** and select **Restore Database**. The **Restore Database** window is displayed.
3. Click the **To database** drop-down list, select a name and append \_or for the ORB. For example, retail\_or.
4. Select the **From device** option and click the ellipse button. The Specify Backup window is displayed.
5. Click **Add**, navigate to the directory where you extracted the ORB backup file, select the **orb\_{appropriate version}.bak** file and click **OK**. The Restore Database window is displayed.
6. Select the **Restore** check box next to the orb backup set.
7. Click **Options** in the Select a page area on the left panel of the window. The Restore Options are displayed.
8. Set the **data** and **log** directory locations and click **OK**.
9. Once the database has been restored navigate to the **Users** object within that database, right-click the waveuser login name and select **Delete**.

**Note:** Newer versions of the applications may not have this user account. This is just a cleanup step so if the account is already removed, continue to the next topic.

## Inject Retail code into the OR database

The Object Repository Database is used by the Web and App servers to access application code. The .EXE file in the build folder contains the new object code to be injected into OR Database.

It is not recommended to backup the OR database. If the database becomes corrupt, restore the clean version of the OR and inject the necessary versions of the code.

**Note:** When performing a multiple step database upgrade, you do not need to inject the code of each intermediate version, only the final version of code to be deployed. Because the OR can contain multiple versions of application code, a single OR database can support multiple environments each running a different application version.

1. Launch the application **.EXE** file. The EXE file is a self-extracting archive that will place the ORB files into the same directory as the archive.
2. Double-click the **readme.bat** file to open a command shell.
3. Follow the instructions in the command shell and execute the **ImportBuild.js** script. For example, `ImportBuild.js servername retail_or retaildbo password`
4. Close the command shell window when the script completes. The files created when the files were extracted have been removed. A new file **ImportBuild.log** was created which contains a record of all the changes made to the OR database.

## Create Retail databases using the Retail empty databases

Each major release of the application contains a set of empty databases. The databases are used to create new environments and later, to upgrade existing environments to new releases. Service Packs, Maintenance Releases and Hot Fixes do not always contain a set of empties.

You need to restore the empty databases as the new environment databases.

1. Unzip the file called {build number}\_Release.zip to the same location as the \_Release.zip file. The conversion package is unpacked and a new subdirectory is created with the build number in the name.
2. Within this subdirectory locate another zip file with the name empty\_{build number}\_{datetime}.zip. This zip file contains a backup of the empty databases.
3. Unzip the file to the same location. This SQL backup file contains all databases for the new environment, except for the REFS database, which is created by the Generic Deploy.
4. From the databases menu within SQL Server Management Studio right-click and select **Restore Database**. The Restore Database window is displayed.
5. In the **To database** field, enter a database name for the new environment. As described in the Prerequisites, use the environment name such as **retail**. Restore the backup sets as:
  - retail
  - retail\_archive
  - retail\_archive\_yyyy\_mm
  - retail\_wh

- retail\_pos
6. Select the **From device** option and click the ellipse button. The Specify Backup window is displayed.
  7. Click **Add**, navigate to the BAK file which contains the empty database you want to restore and click **OK**. The Specify Backup window is displayed.
  8. Click **OK**. The Restore Database window is displayed.
  9. Select the **Restore** check box next to the matching backup set.
  10. Click **Options** in the Select a page area on the left panel of the window. The Restore Options are displayed.

**Note:** You can only restore one backup set at a time.

11. Change the paths for data and log directories as needed and click **OK** to restore the database.
12. Repeat the restore database steps for the remaining 4 databases. When complete, you should have the following databases:
  - Retail
  - Retail\_archive
  - Retail\_archive\_yyyy\_mm
  - Retail\_wh
  - Retail\_pos

## Change database read-only property to False

Empty Retail databases are normally read only. When using the empty databases to create a new environment, you will need to change the Database Read-Only property to False.

1. From the Restore Database window, right-click a database and select **Properties**
2. Click **Options** in the Select a page area on the left panel of the window. The Restore Options are displayed.
3. Scroll down to the State section and set the **Database Read-Only field** to **False**.
4. Repeat to change the read-only property for the remaining 4 databases to False.

## Generic deploy

A Generic Deploy performs several application housekeeping tasks. It may be performed at any time (the environment must be offline) and it **MUST** be performed any time an application database set is restored to a different server or to a different name. The generic deploy is accomplished by running the **ScriptRunnerV2c.exe** file in the Generic Deploy folder, which executes SQL scripts to perform many database related tasks including:

- update cross database name references
- set database compatibility level
- setup/rebuild replication

- create SQL agent jobs
- create current monthly archive database
- create REFS database, if required

**Notes:**

- The Generic Deploy must be run using SQL Login credentials with sysadmin rights such as the Retail Administrator user we created earlier. The Generic Deploy updates the SQL master database and creates specific monthly archive databases.
- ScriptRunner CANNOT execute successfully against an SQL named instance. Attempting to do so will cause parts of the Deploy to NOT execute correctly. If SQL has been installed as a named instance, create a 32 bit SQL Alias to use instead (configurable under SQL Server Configuration Manager/SQL Native Client Configuration).

1. Navigate to the **Generic Deploy** folder and run **ScriptRunnerV2c.exe**.

**Important:** ScriptRunner must be executed locally on the SQL server which contains the client and empty databases.

**Note:** The following includes information when running ScriptRunner:

- **Named Instances:** When SQL Server is installed as a named instance, you will need to create a 32-bit SQL alias. The ScriptRunnerV2 software cannot accept an instance name.
- **Logins:** The login must be an SQL user account with sysadmin rights. ScriptRunnerV2 does not accept windows accounts.

2. In the **Target Database** section, enter the database server name, username with system administrator privileges and password and then click **Connect** to connect to the databases.
3. Click the **Wave Database** drop-down list and select your target database. The Warehouse and Archive database names populate automatically.
4. In the **Replication Info** section enter valid log, data, and temp directories, and then click **Run** to execute pre-scripts. Pre-scripts are used to check the client databases and SQL server for any potential issues prior to upgrade. A warning is displayed asking you to confirm running the package.

**Note:** The pre-scripts do not make any data changes. You may cancel the maintenance after the pre-scripts execute.

5. Click **Yes**. A confirmation is displayed notifying you that the pre-scripts are complete.
6. Click **OK**. The Output files are displayed in a window.
7. **Convert.out** and **scriptrunner.out** are log files. Any other files in the list will contain warnings or errors. Review any errors/warnings and address as needed. If you are not sure about any items here, contact your JDA associate before continuing.
8. Close the window. A confirmation window is displayed asking if you want to continue and an "out" folder is created that contains all out files that you may review later.

**Note:** Each time ScriptRunner is executed, this folder is overwritten. To save the results, rename the folder.

9. Click **Yes**. A confirmation message is displayed that notifies you where the files are located.
10. Click **OK**. The out files are displayed in a window.
11. Review the out files for errors.

**Note:** The 440\_create\_monthly\_archive is expected in new environments and can be ignored; the **verification.sql.out** may or may not appear, but can also be ignored. JDA's Data Services team can review out files for potential errors.

12. Close Scriptrunner.

## Grant retail database user access

1. Open SQL Server Management Studio.
2. Click **Security** and select **Logins**.
3. Right-click the account you created for the application and select **Properties**.
4. Select **User Mapping** and select each of the environment databases (including the refs database) and grant the user **db\_owner** access.
5. Select the ORB database and grant the user **db\_datareader** access and click **OK**.

**Note:** JDA recommends additionally granting access to the read-only user account using the following steps:

- a. Open SQL Server Management Studio.
- b. Click **Security** and select **Logins**.
- c. Create an account for read only access to the application databases. For example, **retail\_ro\_user**
- d. Right-click the account you created for the application and select **Properties**.
- e. Select **User Mapping**, select each of the environment databases and the OR database and grant the user **db\_datareader** access and then click **OK**.
- f. Select each of the environment databases and delete the **waveuser** user.

**Note:** In newer versions, this user may not exist.

6. If CRP will be used in this implementation, create and grant access rights to an SQL user account for CRP. See the CRP Configuration section.

# Chapter 3. Internet Information Server (IIS) configuration

## Summary

This chapter of the installation guide provides information for the initial configure of Internet Information Server (IIS) on a Windows 2016 or 2012R2 server.

## Prerequisites

Windows Server 2016 or 2012 R2 Standard, or Data Center edition installed with the **US Regional settings**.

All the web and application servers within the environment will need to have the following software installed:

Windows Server 2016 and 2012R2	
Software	<p>IIS components detailed in Components needed for Retail (Classic) (on page "<a href="#">Internet Information Server (IIS) configuration</a>" (on page 15))</p> <ul style="list-style-type: none"> <li>• Microsoft Report Viewer Redistributable 2008 Service Pack 1</li> <li>• Microsoft Visual C++ 2005 ATL Update KB973923 (both 32-bit and 64-bit versions are required)</li> <li>• Microsoft Visual C++ 2010 Redistributable (both 32-bit and 64-bit versions are required)</li> <li>• MSXML 6.0 Service Pack 1 with Security Update KB954459</li> <li>• Windows Update (KB2999226) for auto-scheduling</li> </ul>

## Configure IIS

1. Navigate to **Server Manager**.
2. Click **Add roles and features** on the Dashboard page Welcome tile or select **Manage > Add Roles and Features**. The Add Roles and Features wizard is displayed.
3. If the Before You Begin window is displayed, review the information and then click **Next**. The Select installation type window is displayed.
4. Select **Role-based** or **Feature-based** installation and then click **Next**. The Select destination server window is displayed.
5. Choose the **Select a server** from the server pool radio button, select your server and click **Next**. The Server Roles page is displayed.
6. Select **Web Server (IIS)**.
  - a. Locate Web Server (IIS) from the list of roles.
  - b. Expand that selection to get the Web Server selection.

c. Then expand the web server selection and select the following check boxes in each section.

- Common HTTP features
  - Default Document
  - Directory Browsing
  - HTTP Errors
  - Static Content
- Health and Diagnostics
  - HTTP Logging
  - ODBC Logging
  - Request Monitor
- Performance
  - Static Content Compression
  - Dynamic Content Compression
- Security
  - Request Filtering
  - Basic Authentication
  - Windows Authentication
- Application Development
  - .NET Extensibility 3.5
  - .NET Extensibility 4.5
  - ASP
  - ASP.NET 3.5
  - ASP.NET 4.5
  - ISAPI Extensions
  - ISAPI Filters
- Management Tools
  - IIS Management Console
  - Management Service
  - IIS 6 Management Compatibility



**Note:** Selecting this check box automatically selects all the check boxes for the IIS 6 Management Compatibility sub-elements.

- IIS Management Scripts and Tools
  - Management Service
7. Click **Next** and then click **Install**.
  8. Click **Next**. The Confirm Installation selection window is displayed.
  9. Review your selections and click **Install**. A message may ask whether you want to restart the server.
  10. To restart the server now, click **Yes**. You may need to close other applications manually before you can restart. After the server restarts and you log in, the installation completes and the Installation Results window is displayed.

**Note:** To restart the server later, click **No**. The changes are not applied until you restart the server.

11. Confirm that the installation has succeeded and then click **Close**.
12. Click **Next**.
13. Click **Install**.
14. When the installation is complete, click **Close**.
15. Close Server Manager.
16. Navigate to **Start Administrative Tools Internet Information Services (IIS) Manager**.
17. Click the server name.
18. Double-click **logging**. The Logging window is displayed.
19. Click **Select Fields** and select all check boxes.
20. In the **Directory** field, enter a non-system drive.
21. Select the **Use local time** for file naming and rollover check box.
22. Click **Apply**.
23. Double-click **ASP**. The ASP window is displayed.
24. Select the following settings:
  - **Enable Parent Paths: True**
  - **Response Buffering Limit: 41943040**
  - **Script Time-out: 00:05:00**
  - **Enable Session state: False**
25. Click **Apply**.

26. Double-click **Authentication**. The Authentication window is displayed.
27. Perform one of the following for IIS Anonymous Account:
  - If you have single server environments, leave the default setting.
  - If you have multiple server environments, use the same account on all servers. A domain account is recommended. If you are using local accounts, verify the account name and password is the same on all servers.

28. To configure an account for Anonymous Access:
  - a. Open the **Authentication** settings.
  - b. Click **Anonymous Authentication**.
  - c. In the **Actions** pane, click **Edit**.
  - d. Click the **Set** button and enter the credentials for this account.
  - e. Click **OK** twice.

**Note:** Add this user to the **IIS\_IUSRS** group in **Local Users and Groups**.

29. Perform this step only to utilize a custom account for the Application pool, otherwise the recommended default will be used (ApplicationPoolIdentify)
  - a. Click **Application Pools** in the left pane.
  - b. In the Actions pane, click **DefaultAppPool** and **Advanced Settings**. The Advanced Settings window is displayed.
  - c. In the Process Model section, click **Identity** and then the ellipse button. The Application Pool Identity window is displayed.
  - d. Select **Custom account** and click **Set**. The Set Credentials window is displayed.
  - e. Enter the account credentials and click **OK** twice.
  - f. Click the left arrow button.
30. Navigate to **Administrative Tools > Component Services** in Control Panel.
31. Expand **Component Services Computers My Computer Distributed Transactions**.
32. Right-click **Local DTC** and select **Properties Security**.
33. Select the following:
  - Network DTC Access
  - Allow Remote Administration
  - Allow Inbound
  - Allow Outbound
  - No Authentication Required

- Enable XA Transactions
34. Click **OK**. A message is displayed notifying you that the MS DTC service will be stopped and restarted.
  35. Click **Yes** to proceed.
  36. Click **OK**.
  37. Close Component Services.
  38. If the server is a member of a cluster:
    - a. Move the MSDTC resource to the passive node of the cluster.
    - b. Use the instructions in this document verify that the security settings are the same. If they are not, correct them.
    - c. Move the MSDTC resource back to the original node.
    - d. Take the SQL Server resources off-line and bring it back on-line.

**Notes:**

- It is recommended to restart DTC when changes are made to its configuration.
- EP requires the HTTP request URL and "Referer" header to use the same protocol, whether HTTP or HTTPS. If SSL/TLS offloading occurs prior to the request reaching the EP origin server (such as in a load balancer), then the Referer header must be rewritten to HTTP there. If every node from the browser to IIS runs in HTTP or HTTPS, then the header will not require a rewrite.

# Chapter 4. Web or application deployment

## Prerequisites

- Windows Server installed per above.
- Application Specific Pre-Requisites
- Server must be able to access a local HM Deploy Server
  - Local Host Management Server: requires an Active Directory account with administrative rights on the Web/App servers you are deploying and an HM user account with the Host Management role
- When using the Enterprise Deploy Tool, the HM Deploy URL needs to be added to the Local Intranet Security Zone in Internet Explorer on the target server.

## Files needed

**globalsettings.xml**

**LogConfig.xml**

## Windows Server 2012

**Microsoft .NET Framework 1.1 SP1:** This component should be included in the Release Notes for the applicable versions of the applications. It is included in the basic installation of Windows Server 2003 so it was not listed as a prerequisite. Microsoft did not include it in Windows Server 2012.

**Important:** .NET 1.1 is required for Enterprise Store Operations, but not for Workforce Management for Retail or an HM deploy server.

## Database connection authentication

Retail applications support Native (SQL Authentication) and Trusted (Integrated Authentication) SQL Authentication. This document uses SQL Authentication in the examples. In this mode, all access to the databases by a user of the UI or by the APE service uses the supplied SQL account. The SQL account requires "dbowner" rights to the environment databases and "datareader" rights to the OR database. Users do not have direct access to the databases. When properly logged into the application, access to the database is only through the UI and limited by the permissions granted to their application account.

## Environment naming

A Retail environment is composed of a set of databases, a web site, a URL and a namespace. Choosing an environment name might seem trivial, but it drives the naming of all the components and is not easily changed later. The driving factor for the complexity is not in the naming itself, but in the confusion that can occur when a naming convention is not followed and multiple environments exist at the same location. At this point in the installation, the databases should have been created and an environment name selected. But the database name is the only component so far and its name can still be changed. As this

procedure is accomplished, the other components will be created and deployed making a name change more complex. If you do not know the environment name, check the name of the database set you will be using. All databases in the set start with the environment name. In this example, the project is JDA and this is the Production environment so the environment name is retail and the databases set could be as follows:

- retail\_XXX
- retail\_XXX\_archive retail\_XXX\_archive\_YYYY\_MM retail\_XXX\_wh retail\_XXX\_pos

Naming conventions can be determined by the customer, as long as the DB suffixes are maintained.

## Object repositories

The Object Repository (OR or ORB) is addressed in the Initial DB Deployment chapter, but it is here that the OR will be referenced by the application and therefore worth noting. The OR database stores reusable code. The OR can contain code for multiple versions of the application, even for different applications. Technically the OR is a component of an environment, but it is a shared component and therefore does not need to contain the environment name in the database name unless you wish to dedicate an OR to a specific environment. Most DB servers will have a single OR database, but multiple environment database sets may exist supporting multiple environments. Generally, the OR is named by project only, such as "retail\_or".

## Enterprise Deploy Tool notes

There are 2 issues that have come up that can be resolved by adding an entry to the Target Server's host file that maps support.bluecube.com to the IP address of the local HM (Hosting Management) Deploy Server.

The Enterprise Deploy Tool contains a hard coded reference to http://support.bluecube.com. There is nothing in the deploy process that needs to use this specific site, just a valid HM Deploy Server is required.

When installing the tool from a local HM Deploy Server, you may see an error that reports the discrepancy and the install fails. Install the tool from your local HM Deploy Server. Once the tool has been installed, you may use the actual reference to the local HM Deploy Server.

## Web or application server initial deployment

1. Select the drive to be used for the application and create a directory for the web site. It is recommended that a non-system drive be used. For example, a good directory to use might be **D:\Retail**. The deploy tool will automatically generate the necessary file structure within this chosen location. You will place your **GlobalSettings.xml** and **logconfig.xml** files here.
2. Create five application data folders within the selected drive. For example, **\Retail\archive**, **\Retail\AsynchQueueStore**, **\Retail\ImportExport**, **\Retail\logs**, **\Retail\xmlfiles**.

**Note:** These locations will be entered in the **globalsettings.xml** and **logconfig.xml** files.

3. Grant **Modify** access for the **IIS anonymous user account** and the **APE Service account** to the Web site and Data folders.
  - a. In the file explorer navigate to one of the data folders, select the folder, right-click & go to **Properties**. The properties window is displayed.

- b. Select the **Security** tab. Under the Group or user names box, click **Edit**. The Permissions window is displayed.
  - c. Under Group or user names click **Add**. The Select Users, Computers, Service Accounts, or Groups window is displayed.
  - d. Next to the **From this location** field, click **Locations** and select your machine name (the root of the tree structure). Click **OK**.
  - e. In the **Enter the object names to select** field, enter "IUSR" and click **Check Names**. If done correctly, IUSR should now be underlined in the text area. Click **OK** to close the window. The Permissions window is now displayed.
  - f. In the Permissions window, select the IUSR user name and check the **Modify** check-box for **Allow** under Permissions for IUSR. Click **Apply** and then **OK** to close the window. The Properties window is now displayed.
  - g. Click **OK** to close the window.
4. At a minimum, share the following data folders and Grant **Change/Contributer** access to **Everyone** to each of the data folders:
    - **\Retail\AsynchQueueStore**
    - **\Retail\ImportExport**
    - **\Retail\Xmlfiles**
  5. Configure **GlobalSettings.xml** located in **[drive letter]:\Retail** folder.

**Note:** Any attribute not listed in the following sections should maintain the default value shown in the sample **GlobalSettings.xml** file. There are some XML attributes not listed in this document that may be specific to your installation. This document covers the primary attributes you will use to install Enterprise on a web or application server.

- a. Enter the server name of the SQL Server which houses the ORB and client databases.

**XML Attributes:**

```
GlobalAppSettings ObjectRepository="server={server name}"
Namespace WaveDatabaseConnection= "server={server name}"
Namespace WaveDatamartConnection= "server={server name}"
Namespace WaveArchiveConnection= "server={server name}"
Namespace WaveWebposConnection= "server={server name}"
```

**Note:** This may be a NetBIOS name or FQDN.

- b. Update the **ORB database** name.

**XML Attribute:**

```
GlobalAppSettings ObjectRepository="database={database name}"
```

- c. Update the names of the **client databases**.

**XML Attributes:**

```

Namespace WaveDatabaseConnection= "database={database name}"
Namespace WaveDatamartConnection= "database={database name}"
Namespace WaveArchiveConnection= "database={database name}"
Namespace WaveWebposConnection= "database={database name}"

```

- d. Enter the **SQL ID** and **password** for the DBO account used to manage the client and ORB databases.

**XML Attributes:**

```

GlobalAppSettings ObjectRepository= "uid={SQL ID};pwd={password}"
Namespace WaveDatabaseConnection= "uid={SQL ID};pwd={password}"
Namespace WaveDatamartConnection= "uid={SQL ID};pwd={password}"
Namespace WaveArchiveConnection= "uid={SQL ID};pwd={password}"
Namespace WaveWebposConnection= "uid={SQL ID};pwd={password}"

```

- e. Enter the path to **ImportExport** directory and share.

**XML Attributes:**

```

GlobalAppSettings __ImportExportDirectory="{local path}"
Namespace __ImportExportDirectory="{UNC Share}"

```

- f. Define the **NameSpace name**. This is going to be the instance name of the EP environment.

**XML Attribute:**

```

Namespace name="{name}"

```

**Note:** The NameSpace name may be any name you choose but it is generally a best practice to make the same name as the IIS virtual directory which will be hosting the environment.

- g. Set the **ResolvedHostName** for the namespace.

**XML Attribute:**

```

Namespace __ResolvedHostName= "{host name}"

```

**Note:** This may be a NetBIOS name or FQDN. This value should be local machine name.

- h. Set the **ImportWebServerURL** value.

**XML Attribute:**

```
NameSpace __ImportWebServerURL= "http://{server name}"
```

**Note:** This may be a NetBIOS name or FQDN. This value should be local machine name.

- i. Set the **POSAppServerURL** value.

**XML Attribute:**

```
NameSpace POSAppServerURL= "http://{server name}"
```

**Note:** This may be a NetBIOS name or FQDN. This value should be local machine name.

- j. Set the **PDFServerURL** value.

**XML Attribute:**

```
NameSpace __PDFServerURL= "http://{server name}:{port number}"
```

**Note:** This may be any server, or number of servers, which host the PDF Server components. If multiple servers are listed they should be separated by a comma. This may be a NetBIOS name or FQDN.

- k. Set the **POSFileShareRoot** value.

**XML Attribute:**

```
NameSpace __POSFileShareRoot= "\\{server name}\xmlfiles"
```

**Note:** This attribute is only used if a POS system is being used with Enterprise. This may be a NetBIOS name or FQDN.

- l. Set the **SolverServerAddresses** value.

**XML Attribute:**

```
NameSpace __SolverServerAddresses= "{127.0.0.1}:{16000}"
```

**Note:** The default port number is 16000, unless a different port was selected in the Retail Config Tool.

- m. Set the **TaskQueueStore** value.

**XML Attribute:**

```
NameSpace __TaskQueueStore="[drive letter]:\EPS\archive"
```



- n. Set the **Protocol** value.

**XML Attribute:**

```
NameSpace __Protocol="http"
```

**Note:** This value may be http or https depending on whether SSL is to be used with the environment.

- o. Set the **WaveDebug** value.

**XML Attribute:**

```
NameSpace __WaveDebug="off"
```

**Note:** WaveDebug is generally off for production environments. It may be turned on for troubleshooting or in QE environments. The valid values are on and off.

- p. Set the **ReportCellLimit** value.

**XML Attribute:**

```
NameSpace __ReportCellLimit="{integer}"
```

**Note:** The default recommended value is 3600. If value set to 0 all reports will render as HTML and bypass PDF rendering.

6. Define the APE Service settings. This is done in the APEService NameSpace. Set the **inheritsfrom** value within the APEService namespace.

**XML Attribute:**

```
NameSpace name="APEService" inheritsfrom="{environment namespace name}"
```

**Note:** The APEService namespace should only be present on application servers.

7. Define the APEService Rules. Please contact JDA for the APE Queue Segmentation whitepaper.

8. Configure **logconfig.xml**. This file should exist in the same application directory defined for **Globalsetting.xml**.

- a. Set the **id value** for each appender.

**XML Attribute:**

```
{appender type} id="{name}"
```

- b. Set the **base\_file\_name** value for all appender types.

**XML Attribute:**

```
{appender type} base_file_name="[drive letter]:\EPS\log\{appender id value}"
```

- c. Set the `roll_size` and `roll_count` for each appender type.

**XML Attributes:**

```
{appender type} roll_size="{integer}" roll_count="{integer}"
```

**Note:** `roll_size` is defined in KB. The default values are typically:

```
roll_size="10000000"
```

```
roll_count="10"
```

- d. Define the rule levels for each appender type.

**XML Attributes:**

```
Rule level="{level value}" appender="{appender type}"
```

## Web site creation

1. Open **Internet Information Services (IIS) Manager**. The Internet Information Services (IIS) Manager window is displayed.
2. Expand the Server node, right-click **Sites** and click **Add Web Site**. The Add Web Site Window is displayed.
3. In the **Site Name** field, enter the environment name for the site. It shares the name of the instance of the EP deploy. The site name must match the name defined in **globalsttings.xml**.
4. In the **Physical path** field, enter the drive:\directory created earlier for the web site.
5. In the **IP Address** field, normally web servers are dedicated to the application and the IP Address can be left to **All Unassigned**.
6. In the **Host Name** field, enter the URL for this web site. If this is left blank, the server responds to any URL that resolves to the server.
7. Deselect the **Start Web** site immediately check box.
8. Click **OK**.

## Application Deploy - Host Management Server

If the Enterprise Deploy Tool will be used instead, see "[Local Deploy – Enterprise Deploy Tool](#)" (on page 29)

1. Connect to the Host Management server. Hosting Management servers are designed to use single sign-on authentication. If yours does not automatically log you in, return to the Hosting Management Installation Document.

**Note:** If the environment you want to deploy has not yet been entered into the Hosting Management system, you will need to enter it at this time. In order for a deployment to be successful, you must have configured the cluster, environment and all of the information for the servers.

2. In Enterprise, click **Hosting Management > Cluster Setup**. The Cluster window is displayed.

**Note:** A cluster is a collection of web and app servers that run one or more environments. Most customers have a production cluster, plus one or more test clusters. Use the Cluster Setup task to create and maintain clusters.

3. Click **Add Cluster**. The Add New Cluster window is displayed.
4. In the **Cluster** field, enter a name that describes the cluster. For example, Customer Name Prod or Customer Name Test.
5. Click the **Configuration** drop-down list and select the Enterprise functionality you will be using. It is not critical you make the right selection during cluster setup. During the deployment, the configuration will be corrected, if needed.
6. Click **Save & Close**.
7. Select the **Cluster Setup** task. The Cluster window is displayed.
8. In the **Name Contains** field, enter the part of the cluster name you just created and click **View Results**.
9. Click the cluster name you just created. The Edit Cluster window is displayed.
10. Click the **Servers** tab.
11. Click **Add Server**. The Add Server window is displayed.

**Note:** If the servers for this cluster have been added already in the Server Setup menu, click **Add Existing Server** and enter the server name.

12. In the **Server Name** field, enter the name of the DB Server.
13. Click the **Location** drop-down list and select the server location.
14. Select the **DB Server** check box.
15. Click the **IP Addresses** tab.
16. Click **Ping for IP Address**. A confirmation message is displayed asking you to save before proceeding.
17. Click **OK**.
18. Click the **Databases** tab.
19. Click **Add Database**. The Add Database window is displayed.
20. In the **Database Name** field, enter OLTP.
21. Click the **Type** drop-down list and select **wave**.
22. Click **Save and Close**.
23. Repeat for the following databases:
  - archive

- archive\_yyyy\_mm
- pos
- wh

24. Click **Save & Close**.

25. From the Edit Cluster window, Servers tab, click **Add Server**.

**Note:** If the servers for this cluster have been added already in the Server Setup menu, click **Add Existing Server** and just the server name.

26. In the **Server Name** field, enter the name of a web or app server in the environment.

27. Click the **Location** drop-down list and select the server location.

28. In the **Bluecube Root Directory** field, enter the root directory where **globalsettings.xml** is located. For example, **E:\Retail**.

29. Select the **App Server** and **Web Server** check boxes.

30. Click the **IP Addresses** tab.

31. Click **Ping for IP Address**. A confirmation message is displayed asking you to save before proceeding.

32. Click **OK**.

33. Repeat the previous steps (from the Edit Cluster window) to add servers for all your web and app servers.

34. Click the **Environment** tab.

35. Click **Add Environment**. The Add Environment window is displayed.

36. Click the **Clients** tab,

37. In the **Environment Name** field enter an environment name.

38. Click the **Purpose** drop-down list and select what the cluster will used for. For example, Production, Development, Testing or Demo.

39. Click the **Category** drop-down list and select the type of application. For example, BOS, Food Service, PCS or WFM.

40. Click the **Management** tab.

41. In the **URL** field, enter the URL used by end users to access the environment.

42. In the **App Server URL** field, enter the URL used to access the app server functionality.

43. In the **Website** field, enter the name of the website configured in IIS.

44. In the **Namespace** field, enter the namespace defined in **globalsettings.xml** of the web/app servers.

**Note:** the namespace should be consistent across all web/app servers for the environment.

45. Click the **Databases** tab.
46. Click **Add Database**. The Add Database to Environment window is displayed.
47. Click the **Database Server** drop-down list and select the database server name.
48. Click the **Database** drop-down list and select the database. For example, OLTP, archive, archive\_yyyy\_mm, pos wh and ORB).
49. Click **Save & Add Another**.
50. Repeat until are databases are in the environment.
51. Click **Save & Close**.
52. Click **Save & Close** to save the new environment.
53. Select the **Cluster Setup** task.
54. Search for the cluster that contains the environment you want to deploy and click the cluster name.
55. Click the **Upgrades** tab.
56. Click **Deploy**. The Deploy Cluster window is displayed.
57. Click the **Release** drop-down list and select the release you want to deploy. The build number should automatically populate.
58. Select the **Restart IIS and Ape** automatically check box.
59. Deselect the **Sync Database Code, Indexes, Constraints, and Data** check box.
60. Click **Start Deploy**. A confirmation message is displayed confirming if you want to proceed.
61. If the information in the message matches the environment you want to deploy, click the **Deploy** button. The Deploy Cluster window is displayed and contains additional information regarding the status of the deployment. If everything is configured properly, the Status column updates throughout the deployment. The status column progresses through the following messages:
  - **Not Started:** During this time the deployment is copying **deploy.exe** to the target servers. This file will be located in the BlueCube root directory that is configured for the environment.
  - **In progress:** This message is displayed while the deployment is running on the target servers.
  - **Complete:** This message indicates that the deployment was successful.
62. If the deployment was not successful, click the **Deploy Log** link and look for errors in the deploy log.

## Local Deploy - Enterprise Deploy Tool

An important prerequisite for this deploy tool to work is the Microsoft Report Viewer 2008 SP1 Redistributable.

If the application was deployed using an HM (Hosting Management) server, skip to the Testing topic.

**Note:** Since the target server will attempt to access <http://support.bluecube.com>, you will need to add an entry to the target server's host file that maps [support.bluecube.com](http://support.bluecube.com) to the IP address of the local HM Deploy server you will be using.

1. Update the registry. When using the Enterprise Deploy Tool for the initial install, the default installation path is **C:\wave**. To change this location, modify the Windows registry as follows:
  - a. Run **regedit**.
  - b. Expand **HKEY\_LOCAL\_MACHINE > SOFTWARE**
  - c. Create **new Keys > Radiant Systems > Wave**
  - d. In the Wave key, create two new **String values**:
    - **GlobalSettingsXML**: this defines the installation path
    - **GlobalSettingsFile**: this defines the path to the **logconfig.xml** file.
  - e. Set the value of both to the **full path and name** of the **globalsettings.xml** file
2. Configure browser settings.
  - a. Configure Internet Explorer as per the description provided in the "[Retail \(classic\) web client requirements](#)" (on page 60) of this document.
  - b. Add the URL of the HM Deploy Server into the Local Intranet Sites list.
3. Launch the server welcome page through a browser. This step verifies the target server is able to communicate with the deploy server.
4. Install the Enterprise Deploy Tool.
  - a. Access the tools installation page using the following URL.  
`http://{deployserver}/tools/deploytool/publish.htm`
  - b. Click **Install**. The Application Install security warning is displayed.
  - c. Click **Install**.
  - d. Close the browser. After installation, the Enterprise Deploy Tool automatically launches. To run the tool again, you must log into the target server with the same credentials used to install, otherwise reinstall the tool. The installation creates a Programs group RedPrairie and creates a shortcut for the tool and help file.
5. Log into the HM Deploy Server.
  - a. Enter the **URL** to the HM deploy server.
  - b. Enter the **account credentials** to access this server.
  - c. Click **Login**.
6. Enter application release information.
  - a. In the **Release** field, enter the official release name of the application version to be installed. If the tool can locate the release, the associated build number returns in the **Build** field.

- b. The settings in the **Web Site** and **Namespace** fields should be automatically populated if the tool was able to determine the settings from **globalsettings.xml**.
  - c. If you want to start services later, deselect the **Restart Services** check box.
  - d. If the server will be an application server, select the **Install APE** check box. For web servers, leave deselected.
  - e. If the server will be an application server, select the **Install File Importer Service** check box. For web servers, leave deselected.
7. Click **Deploy**. The deploy process is displayed in the window. It is also written to the **DeployLog.txt** file in the installation directory. A confirmation message displays when complete.

**Note:** If errors occur, such as a missing prerequisite, correct the error and then click **Deploy** again. The deploy process restarts.

8. Click **OK** and close the Enterprise Deploy Tool.
9. If the **Restart Services** check box was not selected, start the following services:
  - IIS
  - APEService
  - RadFileImporterSvc

## Application testing

1. Open **Internet Explorer** and **Browse** to the URL of your web or application server.
2. Verify that the browser has been configured as mentioned in the Retail (classic) web client requirements (on page "[Retail \(classic\) web client requirements](#)" (on page 60)) and that the URL has been added to the Local Intranet Sites list.
3. IIS, by default, does not allow paths to include files in the bin directory (where **ORF.dll** is located). If the **HTTP Error 404.8** is displayed while attempting to access the application Welcome page, modify the **applicationHost.config** file by performing the following:
  - a. Navigate to **%win%\System32\inetsrv\config**.
  - b. Backup **applicationHost.config**.
  - c. **Open** the file with a text editor, such as Notepad.
  - d. Search for **hiddenSegments**.
  - e. Comment out the line **<add segment="bin" />** .
  - f. **Save and Close** the file.
  - g. **Restart IIS**.
  - h. Open the browser and access the page again.

4. Enter the following URL:

```
http://{server name}/portal/content/getwaveversion.asp
```

**Note:** This screen may be used to check settings and determine if the server is able to connect the databases.

5. Scroll to the bottom of the GetWaveVersion screen.
6. Click **Click to run Health Test**. The Health Test window is displayed.
7. In the **Login** field, enter a valid login for the environment.

**Note:** If this is a new environment, there will only be one default login available (waveadmin\admin).

8. In the **Password** field, enter a valid password for the environment.
9. Click **Login**. The system performs four system checks:

- **Application Login:** Login to the application is executed.
- **APE Service Test:** An APE job is sent to the APE Service to test job processing.
- **Database Replication Test:** A SQL database replication test is executed to ensure replication between the OLTP and Warehouse databases is functioning correctly.
- **Database Connection Test:** Connection tests to the OLTP, Warehouse, and ORB databases are performed.

**Note:** In an environment with multiple application servers the APE test job may be processed by any one of the available application servers. To ensure processing on the local server stop the APE service on any other application servers attached to this environment.

## APE Controller support

The APE Controller page can only be accessed by a user with Enterprise Admin authorization. This page displays diagnostic information about the queue segments running on this server. It also allows you to execute a Spin Down, which notifies APE to complete the processing of all current tasks, but not accept any new tasks.

The APE Controller page accesses the APE Services DCOM object and requires that the IIS Anonymous account be granted access to these properties.

1. If the APE service is installed, update its DCOM security settings.
  - a. Navigate to **Administrative Tools > Component Services**.
  - b. Expand **Component Services > Computers > My Computer > DCOM Config**.
  - c. Right click **APEService >** select **Properties**.
  - d. Click the **Security** tab.
  - e. Launch and **Activation Permissions >** select **Customize > Edit**.



- f. Add the **IIS Anonymous User account** for the web site.
  - g. Enable **Local Activation**.
  - h. Click **OK** until returned to the Component Services window.
  - i. Open the **Services Console**, restart **Distributed Transaction Coordinator**, restart the **APEService** service and attempt to access the **APE Controller UI**.
2. If the APE service is installed, configure the Data Execution Prevention.
  - a. Right click on **My Computer** > select **Properties**.
  - b. Click the Advanced system settings.
  - c. Click **Performance Settings**. The Performance Options window is displayed.
  - d. Click the **Data Execution Prevention** tab.
  - e. Select **Turn on DEP for essential Windows programs and services only**.
  - f. Click **OK** twice.

# Chapter 5. Hosting Management server

This chapter of the installation guide presents configuration of the Hosting Management Server (HMS). The HMS can be configured as a single server with both MSSQL and HMS or just HMS with the databases on a shared server.

## Hosting Management Server

- Windows Server must be installed and configured in accordance with the Install documents for IIS
- MSXML 4.0 SP 2
- ADO 2.8
- .NET Framework 3.5 SP 1
- MS Report Viewer Redistributable 2008, Service Pack 1
- Visual C++ 2005 Redistributable Package, Service Pack 1
- Visual C++ 2005 Redistributable Package SP 1 ATL Hotfix
- SQL Client Tools
- SQL Server 2005 Backward Compatibility Components
  - Data Transformation Services 2000 Runtime
  - SQL Distributed Management Objects (SQL-DMO)

## Configuration Requirements

- HMS should be a member of the Active Directory domain. If the HMS is in a workgroup, only local deploys using the Enterprise Deploy Tool can be performed
- HMS should be "enabled for delegation" in Active Directory Users and Computers
- HMS must be able to access a Target Web/API Server
  - Local HMS -> requires an Active Directory account with administrative rights on the Web/App servers you are deploying and an HM user account with the Host Management role
- The local deploy server address needs to be added to the Local Intranet Security Zone in Internet Explorer on the target server.
- Internet Explorer must be configured as per the description provided in the ["Retail \(classic\) web client requirements"](#) (on page 60) of this document.

## Prerequisites

### DB Server

- SQL Server 2012 with Service Pack 2 or SQL Server 2014 with Service Pack 1 must be installed and configured in accordance with RPS SQL Server setup documentation.

**Note:** For installations with SQL Server 2014, the OLTP DB option Compatibility Level must be set to SQL 2012 (110). This is required to address a performance issue in SQL Server.

- SQL Reporting Services → requires IIS to be installed
- SQL Client Tools
- SQL Server 2005 Backward Compatibility Components
  - Data Transformation Services 2000 Runtime
  - SQL Distributed Management Objects (SQL-DMO)

## Retail software files needed

- OR database backup file
- Empty Hosting Management DB
- **Globalsettings.xml** for HMS
- **Logconfig.xml**
- Tools\_x.x.exe (x.x is version)
- **Orbviews.sql**

## Windows Server 2012

**Microsoft .NET Framework 1.1 SP1:** This component should be included in the Release notes for the applicable versions of the applications. Microsoft did not include it in Windows Server 2012.

## Database connection authentication

The install supports Native (SQL Authentication) and Trusted (Integrated Authentication) SQL Authentication. This document uses SQL Authentication in the examples. In this mode, all access to the databases by a user of the UI or by the APE service uses the supplied SQL account. The SQL account requires the dbowner rights to the environment databases and "datareader" rights to the OR database. Users do not have direct access to the databases. When properly logged into the application, access to the database is only through the UI and limited by the permissions granted to their application account.

## No access to <http://support.bluecube.com>

When using the Enterprise Deploy Tool to deploy from a local HM installation, the target Web/App server running the tool will try to connect to <http://support.bluecube.com>. The server will not have access to this address, and it generates an error and fails. The tool does not require access to the site, but due to limitations and security settings, this address is hard coded in the application. Simply update each target Web/App server's Host file with an entry that associates this URL to the local HM server's IP address.

## Database configuration

1. Within SQL Server Management Studio navigate to **Security > Logins**. A SQL login ID must be configured to manage the Hosting Management database.

**Note:** This account is not typically the same as the dbo management account. The accounts are separated to compartmentalize security access.

2. Right-click **Logins** and select **New Login**. The **Login – New** window is displayed.
3. Perform the following on the Login New window:
  - a. In the **New Login name** field, enter a name. For example HMSAdmin.
  - b. Select **SQL Server Authentication**.
  - c. In the **Password** and **Confirm password** fields, enter a password.
  - d. Deselect **Enforce password policy**, **Enforce password expiration** and **User must change password at next Login**.
  - e. Click **OK**. The SQL Server Management Studio window is displayed.
4. Right-click **Databases** and select **Restore database...**The Restore Database window is displayed.
5. Perform the following on the Restore Database window:
  - a. In the **To database** field, enter a database name for the orb. For example, hm\_or.
  - b. Select **From device** in the Source for restore section and click the ellipsis button and select the source backup set. The Specify Backup window is displayed.
  - c. Confirm the **Backup Media** field contains **File** and click **Add**. The Locate Backup File window is displayed.
  - d. Select the BAK file which contains the ORB database you want to restore and click **OK**. The Restore Database window is displayed.
  - e. Select the **Restore** check box for the backup set you added.
  - f. Change the paths for the data and log directories as needed and click **OK**. A confirmation message is displayed notifying you that the database restore is complete.
  - g. Click **OK**.
6. Extract the empty\_hm database BAK file from its compressed file location.
7. Right-click Databases and select **Restore database...**The Restore Database window is displayed.
8. Perform the following on the Restore Database window:
  - a. In the **To database** field, enter a database name for the HM. For example, HM.
  - b. Select **From device** in the Source for restore section and click the ellipsis button and select the source backup set. The Specify Backup window is displayed.
  - c. Confirm the **Backup Media** field contains **File** and click **Add**. The Locate Backup File window is displayed.
  - d. Select the BAK file which contains the HM database you want to restore and click **OK**. The Restore Database window is displayed.
  - e. Select the **Restore** check box for the backup set you added.

- f. Change the paths for the data and log directories as needed and click **OK**. A confirmation message is displayed notifying you that the database restore is complete.
  - g. Click **OK**.
9. Copy **tools\_x.x.exe** to a directory on the server and run the executable to extract the database code.
10. Double-click **readme.bat**. The command window is displayed.
11. Enter the following command line:
 

```
Tools>importbuild.js virtual-dtl hm_orb sa wave
```
12. Double-click **orbviews.sql** to manually adjust the views in the HM database.
13. Select **Edit > Find and Replace**. The Find and Replace window is displayed.
14. In the **Find what** field, enter %orb%, in the **Replace with** field enter the name of your orb database (hm\_or) and click **Replace All**.
15. Execute the script on the HM database.
16. Configure the HM user account. The SQL login ID must be configured to access the HM and HM\_orb databases.
  - a. Open SQL Server Management Studio Security.
  - b. Select **Security > Logins**.
  - c. Right-click the user account and select **Properties**. The Login Properties window is displayed.
  - d. Select **User Mapping** from the left pane.
  - e. Select the Map check box for the **HM** database and select the **db\_owner** in the role membership section at the bottom of the page.
  - f. Select the Map check box for the **HM\_orb** and select the **db\_datareader** in the role membership section at the bottom of the page.
  - g. Click **Ok**.
17. Start SQL Server Agent.
  - a. Click **Start > Run > Services.msc**.
  - b. Find SQL Server Agent (MSSQLSERVER)
  - c. Verify the service is started and set to Automatic startup.

## Web server configuration

1. Select the drive to be used for the application and create a directory for the web site. It is recommended that a non-system drive be used. You will place your **Globalsettings.xml** and **logconfig.xml** files here. The following structure is recommended:

- **Application:** \HMS
  - **Log4Rad:** \LogFiles\Log4Rad
  - **Web Site Logs:** \LogFiles\IIS
2. Grant **Modify** access for the **IIS anonymous user account** and **the APE Service account** to the Web site and Log folders:
    - **HMS**
    - **LogFiles**
  3. Save a copy of **HM Globalsettings.xml** and **logconfig.xml** in the directory that was created in the previous step.
  4. **Open Globalsettings.xml** for editing.
  5. Modify the following
 

```
[hm db server name]
[hm server name]
[Orb db name] - hm_orb
[orb user name] - hmadmin
[orb user password]
[fully qualified domain name of hm server]
[hm users' password]
[hm db name] - hm
[hm db user name] - hmadmin
[hm db user password]
```
  6. Open **logconfig.xml** for editing.
  7. Edit rules and set directory paths.

## Web site creation

1. Open Internet Information Services (IIS) Manager.
2. Expand the Server node > right-click **Sites** > click **Add Web Site**. The Add Web Site window is displayed.
3. In the **Site name** field enter the environment name for this site.

**Note:** The site name must match the name defined in **GlobalSettings.xml**.

4. In the **Physical path** field, enter the drive:\directory created earlier for the web site.

5. If the site will use SSL and the certificate has been loaded on the server, click the **Type** drop-down list and select **https** and enter 443 in the **Port** field.
6. Normally web servers are dedicated to the application and the **IP address** field can be left to contain **All Unassigned**.
7. In the **Host Name** field, enter the URL for this Web Site. If this is left blank, the server responds to any URL that resolves to the server.
8. Deselect the **Start Web site immediately** check box.
9. Click **OK**.

## Initial HM Deployment

1. Confirm you can access a local HM server for the initial deployment of the new HM server. If not, contact your JDA Services representative for this install.
  - a. Go to <http://support.bluecube.com/tools/deploytool/publish.htm>. The RedPrairie Enterprise Deploy Tool is displayed.
  - b. Click **Install** to begin the installation of the deploy tool (**deploy.exe**) on your local workstation. The File Download – Security Warning window is displayed.
2. Click **Run** or **Install**.
3. If a confirmation is displayed asking you if you want to install the application, click **Install**. The Login window is displayed.

**Note:** For your initial installation, a Login window will not display until you enter the deploy server address in the tool.

4. In the **Login** and **Password** fields, enter **HMDeploy** as the user id and **onetime** as the password.
5. Click **Login**. The Enterprise Deploy Tool is displayed.
6. In the **Hosting Management Server** field, enter the address for the deploy server.
7. Select the **Release Mode** option.
8. In the **Target Directory** field enter the wave directory you created. For example, **c:\wave**.
9. The **Release** field should contain the current release of Tools 3.5.4.6.
10. Click the **Web Site** drop-down list and select the website that HM will be installed under.
11. The **Namespace** field should be populated from your **globalsettings.xml** file.
12. Select the **Restart Services and Install APE** check boxes and click **Deploy**. A confirmation dialog box is displayed when the deploy completes.
13. Click **OK**. The log file is displayed.
14. Check the log for errors.

**Note:** The log file is also located in your root directory (**c:\wave**). It is named **deploylog.txt**

## IIS configuration

1. Open **IIS Manager**.
2. Expand **Server Node** and **Sites**.
3. Click the HM Web site.
4. Double-click **Authentication**. The Authentication window is displayed.
5. Disable Anonymous Authentication for the web site.
  - a. Click **Anonymous Authentication**.
  - b. In the Actions pane on the right, click **Disable**.
  - c. Click **Windows Authentication**.
  - d. In the Actions pane on the right, click **Enable**.
6. Access the Tools virtual directory authentication settings.
  - a. Expand the HM web site and Services
  - b. Click Tools.
  - c. Double-click **Authentication**. The Authentication window is displayed.
7. Enable Anonymous Authentication.
  - a. Click **Anonymous Authentication**.
  - b. In the Actions pane on the right, click **Enable**.
  - c. Click **Windows Authentication**.
  - d. In the Actions pane on the right, click **Disable**.
8. Restart IIS.
  - a. Click **Start > Run > iisreset**.
  - b. Internet Information Manager > start the HM web site.
9. IIS, by default, does not allow paths to include files in the bin directory (where ORF.dll is located). If a HTTP Error 404.8 is displayed while attempting to access the application Welcome page, perform the following to modify **applicationHost.config**:
  - a. Navigate to **%win%\System32\inetsrv\config**.
  - b. Backup **applicationHost.config**.
  - c. Open the file with a text editor such as Notepad.
  - d. Search for hiddenSegments.
  - e. Comment out the line `<add segment="bin" />`
  - f. **Save** and **Close** the file.



- g. Restart IIS.
- h. Open the browser and try the page again.

## Database update

1. Open **orbviews.sql** in a query window pointing at your hm database.
2. Replace **%orb%** with your HM orb database name and execute the script.
3. Extract **one\_time\_db\_server\_deploy.zip** to a temporary directory.
4. Run **ScriptRunnerV2x.exe** from the temporary directory. The Scriptrunner window is displayed.
5. In the Target Database area of the page, enter the server name, system administrator (sa) user and password.
6. Click **Connect**.
7. Click the **Wave Database** drop-down list and select the hm database.
8. Click **Run**. A confirmation message is displayed.
9. Click **Yes**. A confirmation message is displayed.
10. Click **OK**. The Output files from window is displayed.
11. Review the output files for errors.
12. Open **SQL Server Management Studio**.
13. Click SQL Server Agent.
14. Right-click **Jobs** and select **New Job...**The New Job window is displayed.
15. In the **Name** field, enter APE Scheduler.
16. Select **Steps** in the left pane and perform the following:
  - a. Click **New...**
  - b. In the **Step Name** field, enter APE Scheduler.
  - c. In the **Database** field, enter HM.
  - d. In the **Command** field, enter the following:
 

```
declare @current_timestamp datetime
select @current_timestamp = current_timestamp
exec ape_scheduler_clock @current_timestamp
```
  - e. Click **OK**.
17. Select **Schedules** in the left pane and perform the following:
  - a. Click **New...**
  - b. In the **Name** field, enter Every 5 minutes.

- c. Click the **Occurs** drop-down list and select **Daily**.
  - d. Select the **Occurs every** option and enter 5 minutes.
  - e. Click **OK** twice.
18. Login to the Hosting Management Enterprise site using the following credentials:
- Username: waveadmin
  - Password: admin
- Note:** Password admin can be used to change this password.
19. In the Search area enter **Client Setup** to access the Client Setup task. The Client Setup window is displayed.
20. Click **Add Client** and enter information in the following fields:
- **Client Code**
  - **Name**
21. Click the **User** tab and enter information in the following fields:
- **First Name**
  - **Last Name**
  - **Language**
  - **Password** fields
- Note:** This user is the Client Admin User (not the HM User).
22. Click the **Address** tab and enter information in the following fields:
- **Address Line 1**
  - **City**
  - **State**
  - **Postal Code**
  - **Country**
23. Click **Continue**. A confirmation message is displayed.
24. Note the user name assigned on the message and click **OK**.
25. Click the **Language** tab and choose a default language.
26. Click the **Services** tab and select the **Default** and **Hosting Management Explicit** check boxes.
27. Click **Save & Close**.
28. Log out of Enterprise. A confirmation message is displayed.

29. Click **OK**.
30. Launch the Hosting Management Enterprise site again and log in as the Client Admin user you just created.
31. Access the **Employee Setup** task to create a new HM user. The Employee Setup window is displayed.
32. Click **Add Employee**. The Add Employee Window is displayed.
33. Create a new user that will be used to deploy to the web/app servers. This user's login will need to match the user's domain login name. (By Default the user name is the first initial + last name of the user).
34. Enter the appropriate **Name** fields.
35. Click the **Login** tab and enter the information.

**Note:** The password should match the \_\_\_**WaveLoginPassword** value entered in the **globalsettings.xml** file for HM.

36. Deselect the **Force Password change on next login** check box.
37. Select the **Role** tab and select the **Hosting Management** user role.
38. Click **Save & Close**.
39. If the login name does not match the user's domain account you can select the user again and select the **Login** tab and change the **Employee Member ID**.
40. Log out of Hosting Management. You are now ready to log on as the Hosting Management User.
41. Launch the Hosting Management Enterprise site again. This user should automatically log in with your domain credentials.
42. In Enterprise, navigate to the **Hosting Management > Server Setup** task. The Servers window is displayed.
43. Click **Add Server**. The **Add Server** window is displayed.
44. Enter the following information:
  - In the **Server Name** field, enter the name of the HM server.
  - In the **Description** field, enter a server description.
  - Click the **Location** drop-down list and select the location for the server. If no locations exist for the server, select **Add New**, enter a name and description for the new location and click **Save & Close**.

**Note:** A new location is needed for an environment that has a firewall between it and the HM server.

45. Click the **IP Address Tab** and perform the following:
  - a. Click **Ping for IP Address**. A confirmation message is displayed.

- b. Click **OK**.
  - c. Click **Save & Close**.
46. From the command prompt, run the following command replacing the FQDN or hostheader of the HM server and server name.

### **Setspn – A http/servername.domain.com servername**

A registering ServicePrincipalNames will respond with the following:

```
Registering ServicePrincipalNames for
CN=servername,CN=Computers,DC=domain,DC=domain,DC=com
    http/servername.domain.com
```

Updated object

**Note:** The command is NOT http://\*\*

## Troubleshooting HM install

- Verify the install user is an administrator of the box you are installing to.
- Verify the HM web site is in the Local Intranet Zone of the Internet Explorer browser you are using.
- Verify the HM server is in the same domain as the target servers or trusts exist between domains.
- Use the WMI and Copy test page to make sure you have the network configured properly and impersonation is working properly.
  - `http://hm.domain.com/testwmi.asp?hostname=targetserver.domain.com`
  - `http://hm.domain.com/testcopy.asp?hostname=targetserver.domain.com`
    - Use these pages from the HM server and target server. If it works on the HM server and not the target server, you will be able to deploy from the HM server. To be able to deploy from other boxes, check your setspn, AD delegation for the HM server, and domain trusts.

## Globalsettings.xml for HM server

```
<GlobalAppSettings ObjectRepository="Provider=SQLNCLI11;server=[hm db server
name];database=[orb db name];uid=[orb user name];pwd=[orb user password]"
    ServerMode="tools" Build="willbesetbydeploy" IsQE="y">
  <NameSpace name="hm" __ReportServerURL="http://[hm server
name]/reportserver"
    __ResolvedHostName="[fully qualified domain name of hm server]"
    ToolsSMTPServer="[SMTP server name]"
    ToolsSMTPFromAddress="[e-mail address that SQL Test e-mails will be From]"
    __PortalName="Hosting Management" __WaveTestMode="off"
    __ClientPollInterval="15000" __WaveDebug="off" __WaveErrorTrapping="on"
    __DefaultLanguage="English" __Protocol="http"
```

```

    __WaveAutoLogon="on" __ImportExportDirectory="//[hm server
name]\importexport"
    __WaveSecurityCheck="off">
    <__WaveDatabaseConnection value="provider=SQLNCLI11;server=[hm db server
name];database=[hm db name];uid=[hm db user name];pwd=[hm db user password]"
/>
    <__WaveDevToolsConnection value="provider=SQLNCLI11;server=[hm db server
name];database=[hm db name];uid=[hm db user name];pwd=[hm db user password]"
/>
</NameSpace>
<NameSpace name="APEService" inheritsFrom="hm">
    <Rules>
        <Rule namespace="hm" queues="Default" scanInterval="1000" threads="5" />
    </Rules>
</NameSpace>
</GlobalAppSettings>

```

## ORBViews.sql

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_
Configuration_View]'))
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_Configuration_View]
AS SELECT * FROM %orb%.dbo.CM_Configuration'
ELSE
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_Configuration_View]
AS SELECT * FROM %orb%.dbo.CM_Configuration'
GO

IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_
Build_View]'))
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_Build_View] AS SELECT
* FROM %orb%.dbo.CM_Build'
ELSE
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_Build_View] AS SELECT
* FROM %orb%.dbo.CM_Build'
GO

```

```

IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_
Configuration_Release_View]'))
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_Configuration_
Release_View] AS SELECT * FROM %orb%.dbo.CM_Configuration_Release'
ELSE
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_Configuration_
Release_View] AS SELECT * FROM %orb%.dbo.CM_Configuration_Release'
GO

IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_
Moniker_View]'))
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_Moniker_View] AS
SELECT * FROM %orb%.dbo.CM_Moniker'
ELSE
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_Moniker_View] AS
SELECT * FROM %orb%.dbo.CM_Moniker'
GO

IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_
Object_Revision_View]'))
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_Object_Revision_View]
AS SELECT * FROM %orb%.dbo.CM_Object_Revision'
ELSE
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_Object_Revision_View]
AS SELECT * FROM %orb%.dbo.CM_Object_Revision'
GO

IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_
Object_View]'))
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_Object_View] AS
SELECT * FROM %orb%.dbo.CM_Object'
ELSE
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_Object_View] AS
SELECT * FROM %orb%.dbo.CM_Object'
GO

IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_Page_
View]'))
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_Page_View] AS SELECT
* FROM %orb%.dbo.CM_Page'
ELSE

```

```
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_Page_View] AS SELECT
* FROM %orb%.dbo.CM_Page'
```

```
GO
```

```
IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_Released_Version_View]'))
```

```
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_Released_Version_View] AS SELECT * FROM %orb%.dbo.CM_Released_Version'
```

```
ELSE
```

```
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_Released_Version_View] AS SELECT * FROM %orb%.dbo.CM_Released_Version'
```

```
GO
```

```
IF NOT EXISTS (SELECT * FROM sys.views WHERE object_id = OBJECT_ID(N'[CM_User_View]'))
```

```
EXEC sp_executesql @statement = N'CREATE VIEW [dbo].[CM_User_View] AS SELECT
* FROM %orb%.dbo.CM_User'
```

```
ELSE
```

```
EXEC sp_executesql @statement = N'ALTER VIEW [dbo].[CM_User_View] AS SELECT
* FROM %orb%.dbo.CM_User'
```

```
GO
```

# Chapter 6. Hosting management upgrade

This chapter of the installation guide presents upgrading the Hosting Management module.

## Conversionreadme.txt

The **conversionreadme.txt** file will be located in the zip file. Extract the files and drill down to the Conversion folder. The file will normally contain the name that the empty database should be restored as along with any instructions unique to the version of the application being deployed.

## Hosting Management module configuration

### Restore the hosting management empty database

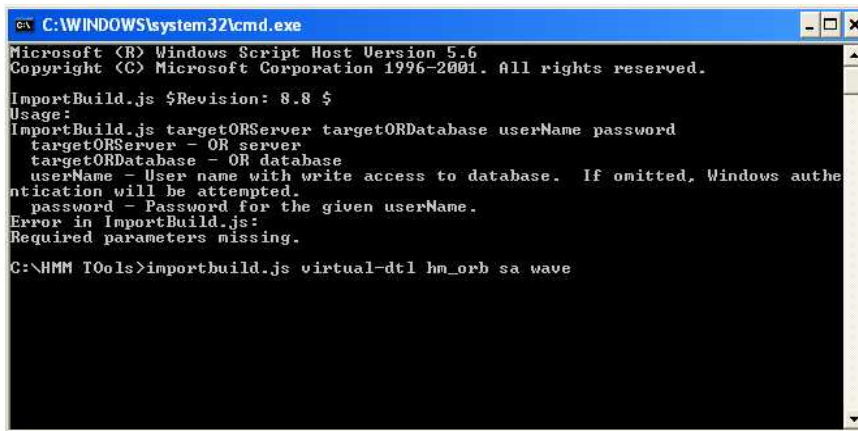
1. Extract the **.ZIP** file.
2. Drill down to the **Conversion** folder.
3. Locate the empty zip file, normally named in the form of **empty\_app\_version\_timestamp.zip**.
4. Extract the database backup file.
5. Start the **SQL Restore Database Wizard**.
  - a. Open **SQL Server Management Console**.
  - b. Connect to the SQL Server hosting the application database.
  - c. Right-click **Databases** and choose **Restore database...**The Restore Database window is displayed.
6. In the **Conversion** folder, review **conversionreadme.txt** to confirm the database name you want to restore. Typically it will use the following form:
 

```
Empty_app_version.
```
7. In the **To database** field enter the name of the database you want to restore.
8. Select the **From** device option, click the ellipsis button to select the source of the backup set. The Specify Backup window is displayed.
9. Click the **Backup Media** drop-down list, select **File** and click **Add**. The Locate Backup File window is displayed.
10. Navigate to the **Conversion** folder, select the empty backup file and click **OK**. The Restore Database window is displayed.
11. Select the **Restore** check box next to the backup set name.
12. Click **Options** on the left pane. The Restore Options window is displayed.
13. Change the file names and paths for the data and log directories as needed and click **OK**. A confirmation message is displayed.
14. Click **OK**. The database restore is complete.



## Application code injection

1. Run **application.EXE**. The files are extracted that need to run against the ORB database.
2. Run **readme.bat**. The file executes the command once and displays the command syntax in the Command console.
3. Enter the command line using the following example:



```

C:\WINDOWS\system32\cmd.exe
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.

ImportBuild.js $Revision: 8.8 $
Usage:
ImportBuild.js targetORServer targetORDatabase userName password
targetORServer - OR server
targetORDatabase - OR database
userName - User name with write access to database. If omitted, Windows authentication will be attempted.
password - Password for the given userName.
Error in ImportBuild.js:
Required parameters missing.

C:\HMM Tools>importbuild.js virtual-dtl hm_orb sa wave

```

- **targetORServer:** SQL Server where the ORB used by the application is hosted.
- **targetORDatabase:** The name of the ORB database used by the application.
- **username** and **password** : Optional, required if you windows login does not have access to the ORB database.

## Database conversion

1. Run **ScriptRunnerV2.exe**.
2. In the Code and Data Source section **Server** field, enter the SQL server name.
3. In the Code and Data Source section **Login** field, enter your SQL user with sysadmin rights login name.
4. In the Code and Data Source section **Password** field, enter the password for the user with sysadmin rights.
5. Click the **Right** arrow button to move the information to the **Target Database** section.
6. Click the Wave **Database** drop-down list in the Target Database section and select the application's Wave database.
7. Click **Run**. A confirmation message is displayed.
8. Click **Yes**. A confirmation message is displayed allowing you to open the output viewer.
9. Click **OK**.
10. Review the output files for errors.
11. Review **Conversionreadme.txt** for any errors.

## Update ORB views

Views within the application database reference your ORB database. You must manually adjust the views to reference the ORB used by the application.

1. Open **orbviews.sql**.
2. Replace %orb% with the name of the orb database used by the application.
3. Execute the orbviews script on the application database.

## Web deploy

1. You can run the deploy tool from the same server for the deployment of the HM web and app server code.
  - a. Go to <http://localhost/tools/deploytool/publish.htm>. The RedPrairie Enterprise Deploy Tool is displayed.
  - b. Click **Install** to begin the installation of the deploy tool (**deploy.exe**) on your local server. The File Download – Security Warning window is displayed.
2. Click **Run**. A confirmation is displayed.
3. Click **Run**.
4. If a confirmation is displayed asking you if you want to install the application, click **Install**. The Enterprise Deploy Tool window is displayed.
5. In the **Hosting Management Server** field, enter the address for the local server.
6. When prompted, enter your account credentials for the local server.
7. The **Target Directory** field cannot be edited. It is the root directory for the application installation. If the entry is not correct, you must modify the Windows registry and re-launch the tool. The default is **C:\wave**.
8. In the **Release** field enter the current release of Tools x.x.x. If the release name is found, the **Build** field automatically populates.
9. Verify the web site for the application and change if necessary.
10. The **Namespace** field should be populated from your **globalsettings.xml** file.
11. If you want to restart services, select the **Restart Services** check box. It is optional and does not affect the deploy.
12. If you want to install APE, select the **Install APE** check box. It depends on the application. The Tools applications require APE.
13. If your servers provide import services, select the **Install File Importer Service** check box.
14. Click **Deploy**. A confirmation dialog box is displayed when the deploy completes.
15. Click **OK**. The log file is displayed.

16. Check the log for errors.

**Note:** The log file is also located in your root directory (**c:\wave**). It is named **deploylog.txt**.

17. If errors are reported, resolve and run the deploy again.

## Web site settings

1. Open **Internet Information Services (IIS) Manager**.
2. In the left pane, expand **Web Sites**, right-click the application web site and select **Properties**. The Display Properties window is displayed.
3. Click the **ASP.NET** tab.
4. Verify the Services/Deploy web application is configured with ASP.NET 2.x.
5. Click the **Directory Security** tab.
6. In the Authentication and access control section, click **Edit**. The Authentication Methods window is displayed.
7. Deselect the **Enable Anonymous Access** check box.
8. Select the **Integrated Windows Authentication** check box.
9. Click **OK** twice.
10. Select all dependent sites for inheritance.
11. In the left pane, expand **Services**, right-click **Tools** and select **Properties**. The Display Properties window is displayed.
12. Click the **Directory Security** tab.
13. In the Authentication and access control section, click **Edit**. The Authentication Methods window is displayed.
14. Select the **Enable Anonymous Access** check box.
15. Deselect the **Integrated Windows Authentication** check box.
16. Click **OK** twice.
17. Restart **IIS**.
18. Launch the Hosting Management Enterprise site and log in to test the site.

# Chapter 7. Upgrade: Database Conversion

## Summary

Database Conversion is the process of upgrading the database tier of an existing environment from one version to a higher one. This process updates a set of databases which are as follows:

DB Type	DB Name (oltp, orb can be any name)
<b>Object Repository database</b>	orb
<b>OLTP</b>	retail
<b>Archive</b>	retail_archive
<b>Archive Monthly Template</b>	retail_archive_yyyy_mm
<b>Warehouse</b>	retail_wh
<b>Point Of Sale</b>	retail_pos

## Prerequisites

As a prerequisite to Database Conversion described in this guide, it is assumed that the current database environment exists on a supported SQL Server version. See DB Server prerequisites.

This guide provides information for a database conversion. It is broken down into several sections. Each section starts with brief explanation of what should be accomplished in the section and why.

## Downgrading

There are no conversion options to downgrade a database to an earlier version. To downgrade, restore a database backup taken when the database was at or before the desired version. If an earlier version is restored, apply the database conversion package until reaching the desired version.

## Convert database

### Upgrade files

Initially, you must acquire the desired Retail software version upgrade files. You will also need the latest available clean ORB, Generic Deploy packages. You can obtain these from your JDA contact.

**Note:** If you already have the files, you can skip to the Update the OR Database topic.

Retail Application for each release consists of 2 or 3 files, the Application Code (.EXE file) and the Database Conversion (.zip) and the Retail Installer(.MSI) for WFMR persona functionalities.

Database Maintenance for Generic\_Deploy package to prepare databases for use.

1. Move all of the downloaded files into a consolidated Retail Database folder.
2. Extract each of the zip files to individual folders.
3. Create a folder for each build you deploy as well as for any updated Generic Deploy and/or Generic Replication packages.

## Update the OR database

The Object Repository Database is used by the Web and App servers to access code. The .EXE file in the build folder contains the new object code to be injected into OR Database.

The OR database is not typically backed up. If the database becomes corrupt, simply restore the clean version of the OR and inject the necessary versions of the code.

**Note:** When performing a multiple step database upgrade, you do not need to inject the code of each intermediate version, only the final version of code to be deployed. Because the OR can contain multiple versions of the application code, a single OR database can support multiple application environments each running a different application version.

1. Launch the .EXE file to extract the application software. The EXE file is a self extracting archive that will place the ORB files into the same directory as the archive.
2. Double-click **readme.bat**. the **readme.bat** file window is displayed.
3. Inject the application software into the ORB.
  - a. Follow the instructions in the commence shell and execute the **ImportBuild.js** script.  
For example, **ImportBuild.js servername retail\_or retaildbo password**.
  - b. When the **ImportBuild.js** script completes, close the command shell window. The files created when the files were extracted have been removed. In their place is **ImportBuild.log** which contains a record of all the changes made to the OR database.

## Restore the empty databases

Each major release of the application contains a set of empty databases. The database are used to create new environments and later, to upgrade existing environments to new releases. Service Packs, Maintenance Releases and Hot Fixes do not always contain a set of empties.

1. Unzip the file **{build number}\_Release.zip** to the same location as the **\_Release.zip** file.
2. Extract the application empty databases. A new subdirectory is created with the build number in the name. This subdirectory contains another zip file with the name *empty\_{build number}\_{datetime}.zip*. This zip file contains a backup of the empty databases. Unzip the file to the same location. This SQL backup file contains all 5 databases for the new environment.
3. From SQL Server Management Studio, right-click **Databases** and select **Restore Database....**The Restore Database window is displayed.
4. In the **To database** field enter the database name for the empty OLTP. The Release folder includes a conversion folder with **conversion\_readme.txt**, which has the correct database name to use. For example,

- empty\_Retail\_{current version}
- empty\_Retail\_{current version}\_archive
- empty\_Retail\_{current version}\_archive\_yyyy\_mm
- empty\_Retail\_{current version}\_wh
- empty\_Retail\_{current version}\_pos

5. Select the **From device** option and click the ellipse button. The Specify Backup window is displayed.
6. Click **Add**, navigate to the BAK file that contains the empty databases you want to restore and click **OK** twice. The Restore Database window is displayed.
7. Select the **Restore** check box next to the matching backup set.

**Note:** You can only restore one backup set at a time.

8. Click **Options** in the Select a page area on the left panel of the window. The Restore Options are displayed.
9. Set the data and log directory locations as needed and click **OK**.

**Note:** These empty databases are only used during conversions so they may be placed on low performance drives.

10. Repeat the restore database steps for the remaining empty databases. When complete, you should have the following databases:

- empty\_Retail\_{current version}
- empty\_Retail\_{current version}\_archive
- empty\_Retail\_{current version}\_archive\_YYYY\_MM
- empty\_Retail\_{current version}\_wh
- empty\_Retail\_{current version}\_pos

## Shutdown the application

The Asynchronous Process Engine (APE) and RadFileImporterSvc (RFI) services run on the app tier servers. Internet Information Server (IIS) runs all Web and App servers.

**Note:** APE, RFI and IIS should all be shut down during database conversions.

1. Stop APE and RFI Services.
  - a. From the Services Management Console, click **Start** and **Run**. The Run window is displayed.
  - b. In the **Open** field, enter services.msc and click **OK**.
  - c. Right-click **APEService** and click **Stop**.
  - d. Right-click **RadFileImporterSvc** and click **Stop**.
2. Stop IIS and all Web and App servers.
  - a. Click **Start** and click **Run**. The Run window is displayed.
  - b. In the **Open** field, enter iisreset/stop and click **OK**.
  - c. Repeat on all web and application servers in the application environment.

## Take backup of databases to be upgraded

It is imperative that a backup be taken before a database is upgraded. In the case of incompatible functionality or unrecoverable failure in a database conversion, the primary roll back option is to restore the databases to the last known good state. Create a backup set of the 5 application databases for the environment you will upgrade. An application environment database set includes:

- retail
- retail\_archive
- retail\_archive\_yyyy\_mm
- retail\_pos
- retail\_wh.

## Start database conversion

Perform conversion by executing the **ScriptRunnerV2c.exe** script included with each application package. It is an executable which executes scripts to perform the database upgrade.

1. Navigate to the **Conversion** folder and double-click **ScriptrunnerV2c.exe**.

**Note:** ScriptRunner must be executed locally on the SQL server which contains the client and empty databases.

2. If SQL Server is running as a named instance, you will need to create a 32bit SQL alias. The ScriptRunnerV2 software cannot execute successfully against a named instance.

**Caution:** You must ensure that a backup of your client databases is performed prior to any conversion.

3. In the Code and Data Source section **Server** field, enter the database server name.
4. In the Code and Data Source section **Login** field, enter your SQL user with sysadmin rights login name.

**Note:** The login must be a sql user account with sysadmin rights. ScriptRunnerV2 does not accept windows accounts.

5. In the Code and Data Source section **Password** field, enter the password for the user with sysadmin rights.
6. Click the **Right** arrow button to move the information to the **Target Database** section. The target Warehouse and Archive database names automatically populate.
7. In the Replication Info section, provide valid log, data and temp directories.
8. In the Version info section, click between the words **Current** and **Version**. The field populates with the current version of the database.
9. Validate that the **Pre-Req Version** and **Current Version** fields match. If the database is on a patch or hotfix past the Pre-Req Version, check the **conversion\_readme.txt** for SQL to allow Scriptrunner to recognize the prerequisite release correctly.

10. Click **Run** to begin Pre-script execution. A confirmation message is displayed.

**Note:** Pre-scripts check the client databases and SQL server for any potential issues prior to upgrade. The pre-scripts do not make any data changes. You may cancel the upgrade after the pre-scripts execute.

11. Click **OK**.
12. Review the output logs for errors. **convert.out** and **scriptrunner.out** are log files. Any other files in the list will contain warnings or errors.
13. If any of the out files contain errors, consult your JDA contact.
14. Close the **Log File Viewer**. A confirmation message is displayed.
15. Click **Yes**.

**Caution:** You must ensure that a backup of your application databases is performed prior to any conversion.

16. Once the conversion completes, you have more output files to review. As with the Pre-Scripts, check with your JDA contact regarding any errors.
17. Close the **Output File Viewer**.
18. Close **ScriptRunner**.

**Note:** JDA's Data Services team should review all out files from Production upgrades for any potential errors. Archive the out files and send to your JDA contact for review.



## Chapter 8. Upgrade: web or application server conversion

Code deployment is the process of installing application code and configuring the web sites on each Web and App server in the environment. There are 2 methods of deploying application code. The easiest and most common method uses a Hosting Management server. The second method uses the Remote Deployment tool.

### Hosting management server

A Hosting Management server (HMS) provides several services in addition to application code deployment. These other services are addressed separately. The primary advantage of the HMS is that it can deploy to all servers in the environment simultaneously. To use HMS, you must inject the Retail code into the HMS's Object Repository (OR) as well as the OR on the database server. During deployment, HMS pushes the code from the OR located on the HMS to each Web and App server. When the Web and App services start up, they reference the OR on the database server as needed.

HMS operates in an Active Directory domain. It is dependent on network and security configurations that are not always available or possible. When the HMS server cannot deploy to the servers, use the Remote Deploy tool. When using a browser to access the HMS across firewalls, you may experience a Kerberos access issue related to the double hop across the firewall. When this happens, you will not be able to start a deploy. If you experience this condition, open a terminal session to the HMS and use the server's browser to access the HMS application.

### Remote Deployment tool

The Remote Deployment tool is for optional install on Web and App servers. The tool can be installed from any HMS. However, newer versions may require a minimum HMS version.

**Note:** The OR on the HMS must contain the desired [Retail or WFMR] version.

Once installed, the tool can be used to pull the code from the HMS OR database down to the target server. If the local firewall settings allow port 80 access to the HMS, this method will successfully deploy the application code.

The disadvantage of using the Remote Deployment tool is that you must run it on each Web and App server in the environment. To use the remote Deployment tool, refer to the Web-App Initial Deploy chapter.

### Code deployment using HMS

1. Verify the Computer Management and Services consoles (.mmc) are not open on any of the web or application servers you are upgrading. The deploy process installs services on the servers. Occasionally, an open console blocks the installation process.
2. Open **Internet Explorer**.
3. Enter the **HMS URL**. The Enterprise window is displayed.

**Note:** Most HM servers are designed to use single sign-on authentication. If yours is setup this way, you should be logged in automatically.

4. If the application code version you want to deploy has not yet been entered into HMS, you will need to inject the code in the HMS's OR database at this time. This procedure is the same as the application code injection on the database server.

**Note:** The document assumes the servers, cluster and environment have already been configured. The HMS application has the same Client PC Configuration Requirements as the Enterprise application. See "[Retail \(classic\) required internet Explorer settings](#)" (on page 60)

5. Select the **Hosting Management** folder from the menu.
6. Select the **Cluster Setup** task. The Cluster window is displayed.
7. Search for and select the cluster that contains the environment you want to upgrade. The **Edit Cluster** window is displayed.
8. Once you have located the cluster you want to upgrade, click the **Upgrades** tab and click **Deploy**. The Deploy Cluster window is displayed.
9. Click the **Release** drop-down list and select the release number that you want to deploy. The **Build** field automatically populates.
10. Select the **Restart IIS and APE automatically** check box.
11. Deselect the **Sync Database code, Indexes, Constraints, and Data** check box.
12. Click **Start Deploy**. The Confirm Deploy window is displayed. It contains a comparison of the HMS and GlobalSettings configurations.

**Note:** Newer versions of HMS may include additional options.

13. If a warning is displayed, the HMS and GlobalSettings configurations do not match and you must click **Cancel**, correct the configuration and try again.
14. Click **Deploy** to confirm the deployment. The status of the deployment for each server in the environment is displayed. If everything is configured properly, the **Status** column will be automatically updated. The possible information displayed in the **Status** field:
  - **Not Started:** During this time the deployment is copying a **deploy.exe** file to the target servers. This file will be located in the application root directory that is configured for the environment.
  - **In progress:** This message is displayed while the deployment is running on the target servers.
  - **Complete:** This message indicates that the deployment was successful.
15. If the deployment was not successful, click the **Deploy Log** link and view errors in the deploy log.
16. If a deploy fails on a server and you correct the problem, click **Try Again** to restart the deploy.

## Deployment testing

1. Get Wave Version. This may be used to check settings and determine if the server is able to connect to the databases.
  - a. Open Internet Explorer.

b. Enter the following URL:

```
http://{retail.domain.com}/portal /content/getwaveversion.asp
```

c. Verify the release and database version settings. The following describes the field information:

- **Release:** Contains the application code version currently deployed.
- **Database Version:** Contains the application database version. The database version may only include the base version such as 2013.3 and not hot fixes or client specific releases.

2. Scroll to the bottom of the screen and click the **Click to run Health Test** link. The Health Test window is displayed.
3. In the **Login** field, enter a valid login for the environment.
4. In the **Password** field, enter a valid password for the environment.
5. Click **Login**.

**Note:** If this is a new environment, there is only one default login available:

**Login:** waveadmin

**Password:** admin

6. The health test performs the following four system checks:
  - **Application Login:** Login to the application is executed.
  - **APE Service Test:** An APE job is sent to the APE Service to test job processing.
  - **Note:** In an environment with multiple application servers the APE test job may be processed by any one of the available application servers. To ensure processing on the local server stop the APE service on any other application servers attached to this environment.
  - **Database Replication Test:** A SQL database replication test is executed to ensure replication between the OLTP and Warehouse databases is functioning correctly.
  - **Database Connection Test:** Connection tests to the OLTP, Warehouse, and ORB databases are performed.

# Chapter 9. Retail (classic) web client requirements

- System requirements:
  - Memory: 2 GB Recommended
  - Display: 1024x768 monitor resolution at 96 dpi

**Note:** It is recommended that you do not use the Windows Large fonts or 120 dpi settings.

- Software
  - Microsoft Internet Explorer 11 (compatibility mode only)

**Important:** If you are using Internet Explorer, you must ensure that opening new browser windows does not merge browsing sessions; otherwise, your Retail (Classic) instance may experience issues, such as form view permission errors. To prevent the default behavior of merging browser sessions, take one of the following actions:

- Edit the shortcut that launches Internet Explorer (or add a new shortcut) and add the -nomerge parameter.
- In the **HKEY\_CURRENT\_USER\Software\Microsoft\Internet Explorer** folder of the Windows registry, add a new DWORD key named MainFrameMerging with a value of 0.

- Microsoft Excel 2000 or later, if you want to view exported data in Microsoft Excel
- Adobe Acrobat 6.0 or later, if you want to view PDF reports

## Retail (classic) required internet Explorer settings

**Note:** These settings apply to Retail (classic) only.

- General
  - Settings (under Browsing history):
    - Check for newer versions of stored pages: Automatically
    - Disk space to use: 200 MB
- Security
  - Local intranet zone > Sites:
    - Automatically detect intranet network: Not selected
    - Include all local (intranet) sites not listed in other zones: Selected
    - Include all sites that bypass the proxy: Selected
    - Include all network paths (UNCs): Selected

- Advanced:
  - Add all of your Retail (Classic) websites.
  - Require server verification (https:) for all sites in this zone: Not selected
- Local intranet zone > Custom level > ActiveX controls and plug ins:
  - Download signed ActiveX controls: Selected
  - Download unsigned ActiveX controls: Not selected
  - Initialize and script ActiveX controls not marked as safe for scripting: Not selected
  - Run ActiveX controls and plug-ins: Selected
  - Script ActiveX controls marked safe for scripting: Selected
- Privacy
  - Settings
  - Blocking Level: Medium
- Advanced
  - First-party Cookies: Accept
  - Third-party Cookies: Accept
- Turn on Pop-up Blocker - Selected
- Advanced
  - Browsing
    - Display a notification about every script error: Selected
    - Reuse windows for launching shortcuts (when tabbed browsing is off): Not selected
  - Security
    - Do not save encrypted pages to disk: Not selected

# Chapter 10. Login Security Configuration

## Recommended Security settings

If Native authentication is chosen for JDA Workforce Management, the recommended settings should be implemented.

Navigate to the Company Profile Setup screen in Classic. From this screen multiple configurations are available to manage login security. The recommended minimum configuration fields are:

On the General Tab

- Session Timeout (min)
- Maximum unsuccessful login attempts
- Unsuccessful login block out time (min)

On the Password Rules Tab

- Maximum Password Age (Days)
- Prevent Reuse of Previous # of Passwords (1-30)
- Minimum Password Length
- Password Cannot Contain User ID or Name
- Password Must Contain One Non-Alphabetic Character
- Password Must Contain One Upper-Case Alphabetic Character
- Restricted Words

The use of this configuration will enhance the security of the application login. The values/settings chosen for a customer should reflect that customer's security policies.

# Appendix A. APE queue segmentation

A typical Enterprise environment processes thousands of tasks through the queue each day. These tasks require varying levels of resources and priority. The primary method of meeting these requirements is through queue segmentation.

The standard configuration of the APE queue provides a single "default" segment which processes all background jobs / tasks indiscriminately. Queue segmentation provides the ability to create multiple segments and add granular control to assign monikers and allocate threads to those segments as required.

This chapter provides an overview of queue segmentation benefits and discusses some of the considerations when implementing segmentation. Intimate knowledge of the target environment and the jobs / tasks being run should be obtained prior to segmenting the queue. APE Queue Segmentation is available in version 6.2.6.5 and later. Specific segmentation configurations should always be tested in a lab environment prior to production implementation.

**Note:** The term moniker is used throughout this document to represent queue objects such as jobs and tasks

## Benefits

There are three key benefits of queue segmentation:

- **Throttle job / task concurrency:** This is useful when a high concurrency of specific job / tasks result in resource hogging or instability. This condition can result in database contention and queue backup. It can also snowball and adversely impact all background processing. Segmentation can be deployed to contain specific monikers within a limited number of threads, thereby limiting the concurrency of those monikers.
- **Prioritize jobs / tasks:** Critical system jobs should not be delayed due to a lack of available threads. Segmentation insures they are not delayed by dedicating threads for specific monikers. These threads are constantly polling for their assigned monikers and will process them promptly when they are queued.
- **Partition APE server resources:** APE server resources can be split into functions to optimize job throughput. Through this mechanism, valuable CPU cycles and memory resources can be best utilized. This method is particularly effective in large APE farms where there is a high volume of background and batch jobs.

## Segmentation types

### Primary types

There are two primary types of queue segmentation, isolated and crossover.

- **Isolated:** Isolated segmentation creates resources which are dedicated to specific monikers or groups of monikers. This type of segmentation is useful for providing guaranteed resources for specific jobs / tasks. It can also be utilized to limit resources available to specific monikers, thereby limiting job/task concurrency.

#### **Isolated Segmentation Examples:**

- Segmentation by task
- Segmentation by thread count

- **Crossover:** Crossover segmentation creates a hierarchy of servicing resources which are allocated to monikers based on priority. This type of segmentation is useful for creating a QoS structure for jobs and tasks.

These fundamental types can be applied to produce unique segmentation models to meet specific requirements. This section covers a few example models:

**Crossover Segmentation Examples:**

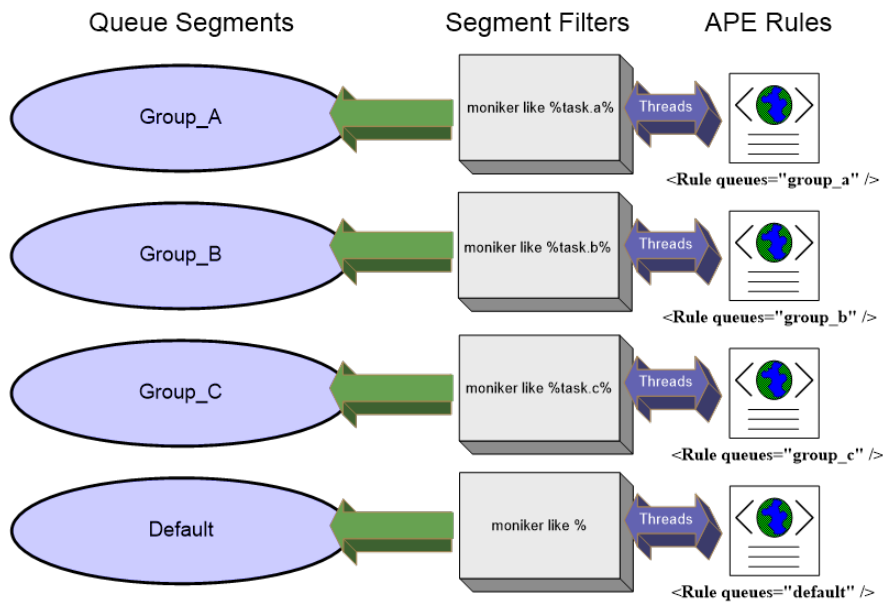
- Segmentation by priority

The characteristics of these models may be combined in segment design. Other models may also be utilized where appropriate.

## Isolated segmentation examples

### Segmentation by task

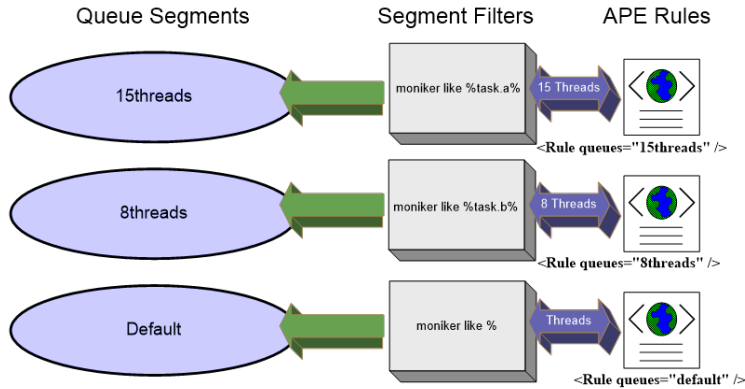
This method of segmentation is the most common. It allocates queue segment(s) to specific tasks. It is useful when there are several different jobs/tasks or groups of jobs/tasks which have unique thread and priority requirements. The following diagram shows the "Segmentation by Task" model.



### Segmentation by thread count

This method of segmentation is useful if there is a requirement to limit the number of threads available to specific monikers. Reducing the concurrency of many jobs can improve performance and stability, while reducing competition for resources at the app and data tiers. The following diagram shows the "Segmentation by Thread Count" model.

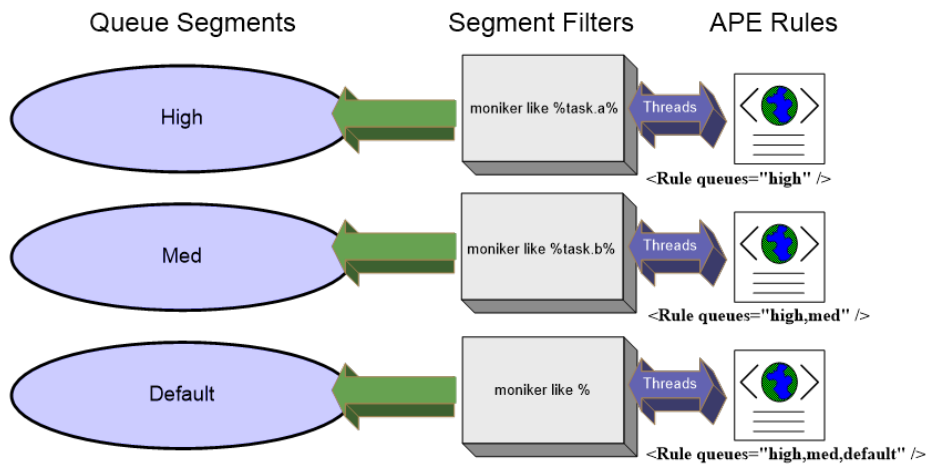




## Crossover segmentation example

### Segmentation by priority

This method of segmentation is useful when there are a small number of jobs/tasks which require a higher execution priority. This type of segmentation applies a low, med, high priority scheme to classify monikers. The following diagram shows the "Segmentation by Priority" model.



Segmentation of an APE queue consists of the following steps:

1. Identify functions requiring segmentation.
2. Determine segment model.
3. Identify monikers and wild card filters.
4. Determine desired rules and thread counts (total threads available).
5. Add queue segments and filters.
6. Add APE rules to **globalsettings.xml** on target servers.
7. Validate segmentation is working properly.

## Identify jobs requiring segmentation

This is the most important step of the segmentation process. An in depth understanding of the background jobs in the environment should be obtained prior to this. Segmentation is not always necessary and could have an adverse effect on system performance. A few questions may help in determining if an environment will benefit from segmentation.

- Do specific jobs cause the queue to get backed up due to high volume or long processing times?
- Do specific jobs cause contention or performance degradation when queued in concurrency?
- Are there mission critical jobs that need RSVP resources?
- Can certain jobs benefit from being spread across APE servers?

If the answer to any of these questions is yes, segmentation should be explored as an option to improve system stability and performance.

## Determine segmentation model

Once it is determined that segmentation is required, a holistic view of all background jobs should be developed and analyzed to determine how to segment. In this step, the number of segments, segment naming conventions and target monikers are identified. As a general rule the number of segments, rules and filters should be kept as low as possible. This keeps the segment design simple and minimizes the risk of introducing problems.

## Identify monikers and wild card filters

After a good understanding of which system functions require segmentation, the target monikers must be identified. One way to do this is by knowing which monikers are going to be segmented and insuring their accuracy by browsing the object repository. An additional method involves looking at the queue history to determine job/task process time and frequency. The APE Status page will provide this information.

1. Log into **Enterprise**.
2. Navigate to **Async processing > APE Status Performance**.
3. Click the **Show queue performance by** drop-down list and select **Moniker**.
4. Use the search options to specify a time period of several weeks.

The moniker is used to determine which segment specific jobs/tasks will end up in. If a moniker does not match a filter, it is simply queued in the default segment. For example, if there was a need to force all punch processing jobs into a dedicated segment, the filter rule may be %punch% while several dozen monikers would match this rule. If the rule should only filter a single specific moniker, the wild card should be expanded as needed. Care should be taken when identifying filter wild cards to insure only the target monikers match the criteria. For example, if the requirement is for the moniker;

**Applications.WorkforceManagement.APE.TAP.tasks.ProcessPunchesForSystem** to go into a specific segment, the wildcard may be; **%TAP.tasks.ProcessPunchesForSystem%**. A job can contain multiple monikers that should be considered when creating filters. Child job and task monikers can be viewed in the "Queued Job Viewer" by drilling into the specific job. If necessary, this information should be confirmed with the appropriate development team.

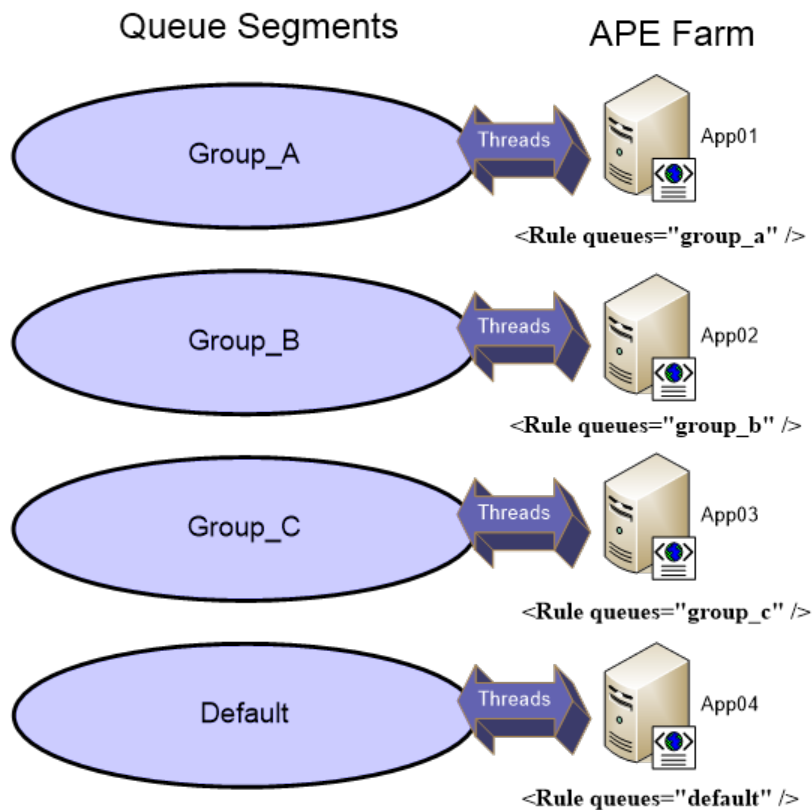
## Determine desired rules and thread counts

Once the segmentation model and target monikers are identified, the APE rules should be created. This is where we specify the number of APE threads available to the segments. Rules added to **globalsettings.xml** are in the following format:

```
<Rules>
  <Rule namespace="%namespace%" queues="%segment%" scanInterval="millisec#"
  threads="thread#" />
</Rules>
```

**Note:** There should always be a rule for the default segment.

You can list multiple comma separated segments in the "queues" element and they will be processed in the respective order by APE in a priority (not round robin) fashion. Therefore, a single rule can service multiple segments. By placing these rules on specific servers, you can partition the load of background processing across the APE farm. The following diagram illustrates this concept.



## Add queue segments and filters

1. Add queue segments. Once the segmentation design is completed, the first step in implementation is to add the segments.
  - a. Log into **Enterprise**.

- b. Navigate to **Async processing > Queue Segments**.
- c. Click **Add New Segment**.
- d. Enter the name of the segment.
- e. Select the **Active** check box and click **Save**.

**Note:** There should always be a default segment.

## 2. Add filters.

- a. Navigate to **Async processing > Queue Segment Config**.
- b. Click **Add New Config**.
- c. Enter the wildcard criteria and select the target segment.

**Note:** Rules should be added in order of job/task frequency with the most frequent at the top. There should always be a rule for the Default segment at the bottom of the list with a global wildcard "%".

## Add APE rules

APE rules should be added to **globalsettings.xml** as presented earlier in the document. The following is an example is from a 6 segment environment that is partitioned by task function.

```
- <NameSpace name="APEService" inheritsFrom="prod">
- <Rules>
  <Rule namespace="prod" queues="labor" scanInterval="20000" threads="3"
  />
  <Rule namespace="prod" queues="ssi" scanInterval="20000" threads="2" />
  <Rule namespace="prod" queues="hr" scanInterval="50000" threads="1" />
  <Rule namespace="prod" queues="punch" scanInterval="10000" threads="1"
  />
  <Rule namespace="prod" queues="rpt-data" scanInterval="10000"
  threads="2" />
  <Rule namespace="prod" queues="default" scanInterval="10000" threads="2"
  />
  <Rule namespace="prod" queues="alert" scanInterval="20000" threads="1"
  />
  </Rules>
</NameSpace>
```

Once the rules have been added, the APE service must be stopped and started for them to take effect. The next step is to validate that the APE service started properly and there is a corresponding **apeengineop.exe** for each rule in global settings (including the default rule).

## Validate segmentation

To insure monikers are being processed in the correct segment you must view the APE execution history.

1. Log into Enterprise.
2. Navigate to **Async processing > Queued Job Viewer**.
3. Use the search options to specify a moniker and time period.
4. When viewing this information insure the required monikers are processed in the correct segment.

Once the segmentation has been validated, continue to monitor job / task throughput and performance for the environment to identify potential issues or opportunities to optimize threads, filters and so on.

## Appendix B. Feature configuration

### Security setup for System Admin

To ensure System Admin user to add/edit users, the security option **Unsorted Employee Form Access** must be enabled. To enable the option, perform the following steps:

1. Log into the Enterprise application using System Admin Role.
2. Navigate to **Feature Setup**.
3. Select the following values:
  - Select the **Feature Name Contains** value as **Unsorted Employee Form Access**.
  - Select the **Type** value as **Security**.
4. Click **View Result**.
5. From the **Roles** tab, select **System Admin**.
6. From the **Services** tab, select **BlueCube**.
7. Click **Save** and then click **Close**.
8. Log out from the application page and re-login to verify.