

IBM WebSphere Enterprise Service Bus



Installing IBM WebSphere Enterprise Service Bus

Version 7 Release 5.1

Note

Before using this information and the product it supports, read the information in “Notices” on page 383.

This edition applies to version 7, release 5, modification 1 of WebSphere Enterprise Service Bus (product number 5724-I82) and to all subsequent releases and modifications until otherwise indicated in new editions.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright IBM Corporation 2007, 2011.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

PDF books and the information center

PDF books are provided as a convenience for printing and offline reading. For the latest information, see the online information center.



As a set, the PDF books contain the same content as the information center.

The PDF documentation is available within a quarter after a major release of the information center, such as Version 6.0 or Version 6.1.

The PDF documentation is updated less frequently than the information center, but more frequently than the Redbooks®. In general, PDF books are updated when enough changes are accumulated for the book.

Links to topics outside a PDF book go to the information center on the Web. Links to targets outside a PDF book are marked by icons that indicate whether the target is a PDF book or a Web page.

Table 1. Icons that prefix links to topics outside this book

| Icon | Description |
|---|---|
|  | <p>A link to a Web page, including a page in the information center.</p> <p>Links to the information center go through an indirection routing service, so that they continue to work even if target topic is moved to a new location.</p> <p>If you want to find a linked page in a local information center, you can search for the link title. Alternatively, you can search for the topic id. If the search results in several topics for different product variants, you can use the search result Group by controls to identify the topic instance that you want to view. For example:</p> <ol style="list-style-type: none">1. Copy the link URL; for example, right-click the link then select Copy link location. For example: <code>http://www14.software.ibm.com/webapp/wsbroker/redirect?version=wbpm620&product=wesb-dist&topic=tins_apply_service</code>2. Copy the topic id after <code>&topic=</code>. For example: <code>tins_apply_service</code>3. In the search field of your local information center, paste the topic id. If you have the documentation feature installed locally, the search result will list the topic. For example: <div><p>1 result(s) found for</p><p>Group by: None Platform Version Product</p><p>Show Summary</p><p>Installing fix packs and refresh packs with the Update Installer</p></div> <ol style="list-style-type: none">4. Click the link in the search result to display the topic. |
|  | <p>A link to a PDF book.</p> |

Contents

| | |
|--------------------------------------|-----|
| PDF books and the information center | iii |
|--------------------------------------|-----|

| | |
|---------|-----|
| Figures | vii |
|---------|-----|

| | |
|--------|----|
| Tables | ix |
|--------|----|

Chapter 1. Installing and configuring

WebSphere ESB

| | |
|--|-----|
| Roadmap for installing and configuring the software | 2 |
| Installing and configuring WebSphere ESB | 2 |
| Planning for WebSphere ESB | 8 |
| Assessing your requirements | 8 |
| Choosing a stand-alone or network deployment environment | 18 |
| Planning your network deployment environment | 21 |
| Planning your database configuration | 35 |
| Planning error prevention and recovery | 61 |
| Preparing to install and configure the software | 67 |
| Preparing operating systems for product installation | 68 |
| Creating the Common database manually before product installation | 74 |
| Installing WebSphere Enterprise Service Bus | 80 |
| Installing the software interactively | 80 |
| Installing WebSphere Enterprise Service Bus silently | 85 |
| Verifying a stand-alone (qesb) installation | 90 |
| Configuring databases | 91 |
| Configuring a Microsoft SQL Server database | 91 |
| Creating the Common database and tables after profile creation or augmentation | 95 |
| Creating database design files by using the database design tool | 97 |
| Creating and configuring the DB2 for z/OS database | 106 |
| Modifying the transaction log options for a DB2 database | 111 |
| Configuring WebSphere Enterprise Service Bus | 111 |
| Creating and augmenting profiles | 112 |
| Configuring databases | 200 |
| Configuring a network deployment environment | 221 |
| Creating and configuring components | 239 |
| Starting the First steps console | 279 |
| Updating WebSphere ESB | 280 |
| Updating the software interactively | 281 |
| Rolling back updates | 282 |
| Manually installing an interim fix | 283 |
| Silently installing an interim fix | 284 |
| Silently uninstalling an interim fix | 285 |
| Uninstalling WebSphere ESB | 286 |
| Uninstalling WebSphere ESB interactively | 286 |

| | |
|-------------------------------------|-----|
| Uninstalling WebSphere ESB silently | 287 |
|-------------------------------------|-----|

Chapter 2. Migrating from earlier products and versions

| | |
|--|-----|
| Migration overview | 289 |
| What is version-to-version migration? | 290 |
| WebSphere ESB Migration roadmap | 291 |
| Migration methods | 293 |
| Migration method comparison | 295 |
| Supported source migration paths | 298 |
| Migration types | 299 |
| Runtime migration tools | 300 |
| Profiles | 302 |
| Mixed-version environments | 303 |
| Databases | 303 |
| Running SQL upgrade scripts | 305 |
| Downtime requirements | 306 |
| What gets migrated | 307 |
| Known compatibility issues | 308 |
| Runtime premigration checklist | 309 |
| Runtime migration procedures | 315 |
| About runtime migration procedures | 315 |
| Migrating a stand-alone environment | 317 |
| Migrating a network deployment environment with full downtime | 322 |
| Migrating a network deployment environment with minimal downtime | 330 |
| Runtime migration subprocedures | 345 |
| Migrating a profile using the profile migration wizard | 345 |
| Migrating a profile using the command-line utilities | 349 |
| Migrating a profile to a remote system | 351 |
| Migrating a server while upgrading an operating system | 353 |
| Migrating databases | 356 |
| Migrating security | 357 |
| Verifying migration | 359 |
| Rolling back your environment | 361 |
| Postmigration tasks | 366 |
| Postmigration tasks for WebSphere ESB | 366 |
| Runtime migration tools reference | 368 |
| Runtime migration troubleshooting | 372 |
| WebSphere ESB deprecated and removed features | 379 |

| | |
|-------|-----|
| Index | 381 |
|-------|-----|

| | |
|---------|-----|
| Notices | 383 |
|---------|-----|

| | |
|-----------------------------------|-----|
| Programming interface information | 385 |
| Trademarks | 385 |

| | |
|------------------------------|-----|
| Sending your comments to IBM | 387 |
|------------------------------|-----|

Figures

- | | | |
|----|---|-----|
| 1. | The three main phases of a product installation: planning, installing, and configuring | 1 |
| 2. | A stand-alone environment | 19 |
| 3. | A network deployment environment | 20 |
| 4. | Remote Messaging topology pattern | 28 |
| 5. | Remote Messaging and Remote Support topology pattern | 29 |
| 6. | Resource allocation example | 30 |
| 7. | Remote Messaging, Support and Web pattern | 31 |
| 8. | Task flow for planning, installing, and configuring the product and the environment | 112 |
| 9. | WebSphere ESB migration roadmap for version-to-version migration | 291 |

Tables

| | | |
|-----|--|-----|
| 1. | Icons that prefix links to topics outside this book | iii |
| 2. | Planning and preparing to install WebSphere ESB. | 3 |
| 3. | Installing to create a stand-alone development environment (qesb) | 4 |
| 4. | Installing to create one or more stand-alone environments | 4 |
| 5. | Installing and configuring WebSphere ESB using the deployment environment wizard | 5 |
| 6. | Installing and configuring WebSphere ESB using the administrative console | 7 |
| 7. | Naming guidelines for nodes, servers, hosts, and cells | 11 |
| 8. | shows the default installation root directory into which the installation program installs both WebSphere ESB and WebSphere Application Server for both root (Administrator) and nonroot users.. . . . | 16 |
| 9. | shows the default installation directory for a profile named <i>profile_name</i> for both root (Administrator) and nonroot users.. . . . | 17 |
| 10. | Installation Manager default installation directories | 17 |
| 11. | Choice of stand-alone or network deployment cluster topology pattern for intended use of WebSphere ESB | 21 |
| 12. | Considerations for selecting a topology for your deployment environment | 33 |
| 13. | Available supplied patterns and their relationship to product features for a distributed (Multiplatform) installation | 34 |
| 14. | Supported database types, their associated dbType values and restrictions | 37 |
| 15. | Databases and their associated <i>feature</i> name. | 38 |
| 16. | Typical stand-alone environment setup | 40 |
| 17. | Typical deployment environment setup | 40 |
| 18. | Supported JDBC drivers and locations that are provided with the product | 41 |
| 19. | Supported JDBC drivers that are not provided with the product | 41 |
| 20. | Database privileges | 43 |
| 21. | Detailed DB2 database privileges | 44 |
| 22. | Detailed DB2 for z/OS database privileges | 45 |
| 23. | Detailed Oracle database privileges | 45 |
| 24. | Detailed SQL Server database privileges | 45 |
| 25. | Scenario: Single user ID or schema | 46 |
| 26. | Scenario 1: Multiple user ID or schema | 47 |
| 27. | Scenario 2: Multiple user ID or schema | 48 |
| 28. | Scenario 3: Multiple user ID or schema | 49 |
| 29. | Databases that are required by individual components | 50 |
| 30. | Supported database products | 51 |
| 31. | Installer options | 52 |
| 32. | Common database script naming convention | 54 |
| 33. | Tables created by WebSphere ESB components | 54 |
| 34. | Supported database products | 56 |
| 35. | Supported database products | 59 |
| 36. | Preparing for installation and configuration | 67 |
| 37. | Applicable database types and their directory names | 75 |
| 38. | DB2 scripts for WebSphere ESB | 75 |
| 39. | DB2 for z/OS scripts for WebSphere ESB | 76 |
| 40. | Oracle scripts for WebSphere ESB | 77 |
| 41. | Default schemas | 78 |
| 42. | Microsoft SQL Server scripts for WebSphere ESB | 79 |
| 43. | Product IDs | 87 |
| 44. | Keys | 87 |
| 45. | | 91 |
| 46. | Selecting the creation option for your stand-alone profile. | 121 |
| 47. | Selecting the profile creation option for your custom profile | 141 |
| 48. | Specified manageprofiles command-line utility parameters | 154 |
| 49. | Defaulted manageprofiles command-line utility parameters | 155 |
| 50. | Specified manageprofiles command-line utility parameters | 156 |
| 51. | Defaulted manageprofiles command-line utility parameters | 156 |
| 52. | Additional manageprofiles command-line utility parameters for Oracle | 157 |
| 53. | Specified manageprofiles command-line utility parameters | 157 |
| 54. | Defaulted manageprofiles command-line utility parameters | 158 |
| 55. | Available manageprofiles parameters for configuration of Common database using DB2 Universal | 159 |
| 56. | Available manageprofiles parameters for configuration of Common database using DB2 Data Server | 160 |
| 57. | Available manageprofiles parameters for configuration of Common database using a database supplied with an i5/OS or IBM i operating system | 161 |
| 58. | Available manageprofiles parameters for configuration of Common database using DB2 for z/OS v8 or DB2 for z/OS v9 | 161 |
| 59. | Available manageprofiles parameters for configuration of Common database using Oracle | 162 |
| 60. | Available manageprofiles parameters for configuration of Common database using Microsoft SQL Server | 164 |
| 61. | Available manageprofiles parameters for configuration of Common Event Infrastructure database using DB2 Universal | 165 |

| | | | | | |
|-----|--|-----|-----|--|-----|
| 62. | Available manageprofiles parameters for configuration of Common Event Infrastructure database using On DB2 Data Server | 166 | 72. | Defaulted manageprofiles command-line utility parameters | 174 |
| 63. | Available manageprofiles parameters for configuration of Common Event Infrastructure database using a database supplied with an i5/OS or IBM i operating system | 167 | 73. | Next step based on whether databases are configured | 180 |
| 64. | Available manageprofiles parameters for configuration of Common Event Infrastructure database using DB2 for z/OS v8 or DB2 for z/OS v9 | 167 | 74. | Database configuration parameters for Profile Management Tool configuration | 195 |
| 65. | Available manageprofiles parameters for configuration of Common Event Infrastructure database using Oracle | 168 | 75. | Required database configuration fields for DB2 Database | 195 |
| 66. | Available manageprofiles parameters for configuration of Common Event Infrastructure database using Microsoft SQL Server. | 169 | 76. | Required database configuration fields for DB2 Universal Database for z/OS. | 196 |
| 67. | Specified manageprofiles command-line utility parameters | 171 | 77. | Required database configuration fields for Microsoft SQL Server | 196 |
| 68. | Defaulted manageprofiles command-line utility parameters | 171 | 78. | Required database configuration fields for Microsoft SQL Server | 198 |
| 69. | Specified manageprofiles command-line utility parameters | 172 | 79. | Required database configuration fields for Oracle | 198 |
| 70. | Defaulted manageprofiles command-line utility parameters | 173 | 80. | Required database configuration fields for Oracle | 199 |
| 71. | Specified manageprofiles command-line utility parameters | 173 | 81. | Required database configuration fields for DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox) | 200 |
| | | | 82. | | 201 |
| | | | 83. | States of a topology instance in order of least to most available | 238 |
| | | | 84. | Event database limitations | 255 |
| | | | 85. | Product IDs | 287 |
| | | | 86. | Version-to-version migration methods: a comparison | 297 |

Chapter 1. Installing and configuring WebSphere ESB

WebSphere® ESB can be installed and configured for multiple topologies. You can install all components on a single server (known as a stand-alone configuration), or you can distribute the components across multiple systems (known as a network deployment configuration). To achieve a highly available environment with failover support, you can install WebSphere ESB into a clustered environment that uses the clustering mechanism of WebSphere Application Server.

About this task

Figure 1 provides a high-level overview of the tasks associated with planning, installing, and configuring WebSphere ESB. The decisions that you make during the planning phase have an effect on the tasks listed under the installing and configuring phases.

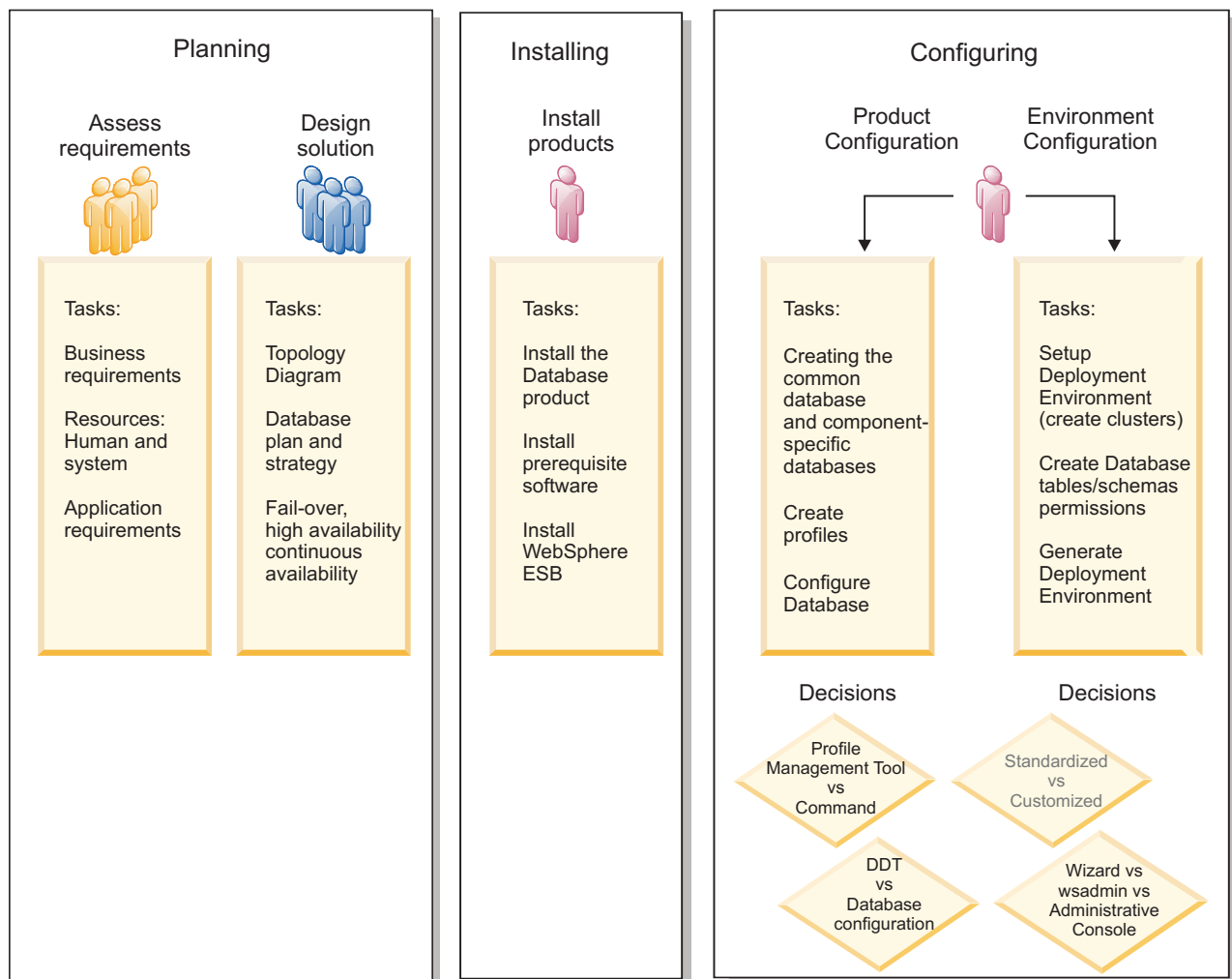


Figure 1. The three main phases of a product installation: planning, installing, and configuring

Related information:



PDF documentation

WebSphere Enterprise Service Bus documentation (in PDF format)



Information roadmaps

Business Process Management information roadmaps on IBM developerWorks organize information about WebSphere ESB, and the other products in the portfolio.



IBM Education Assistant

Multimedia educational modules about WebSphere ESB, provided by IBM Education Assistant.



Technotes

WebSphere ESB Support > Install. Have questions about installing WebSphere ESB? These resources can help lead you through your product installation and setup.



Overview

Overview tab, on product library Web page. Use this page to access announcements, data sheets, and other general library documents related to WebSphere ESB.

Roadmap for installing and configuring the software

The roadmap lists the sequence of tasks that you need to perform to achieve a stand-alone or network deployment environment configuration. Its purpose is to guide you through installing and configuring the software and to help you gain an understanding of the different installation scenarios, so that you can achieve the scenario that best suits your needs.

Installing and configuring WebSphere ESB

Use the tables provided to guide you through the tasks to plan for, prepare for, install, and configure WebSphere ESB for your choice of deployment environment.

Use the following tables as a roadmap, to understand and track what you need to do to install and configure WebSphere ESB. The tables provide links to the detail provided in subsequent topics of this documentation.

Table 2. Planning and preparing to install WebSphere ESB.

| Task | Where to find information | What results |
|---|---|---|
| <p>Plan the environment that you want to create by installing and configuring the software, including the following tasks:</p> <ol style="list-style-type: none"> 1. Choosing to create a stand-alone or network deployment environment 2. For a network deployment environment, choosing to create the topology (of workstations, servers and clusters, and other artifacts) from a standard pattern or your own custom topology 3. Choosing the database setup you want to use 4. Planning other components of your chosen environment. | Planning to install and configure the software | Planning helps you understand the environment that you want to create, prepare a checklist of details for the components to install and configure, and gives you data that you need for the actions you take later. |
| Obtain the software. | WebSphere Enterprise Service Bus | You have obtained the installation media or downloaded the software package from the download site. |
| Prepare to install and configure the software. | Preparing to install and configure the software | Each workstation that you want to use for your chosen environment is ready for you to install and configure the software. |
| Install and configure the software for your chosen deployment environment. | <p>Depending on the environment that you want to create, see one of the following tables in this topic:</p> <ul style="list-style-type: none"> • “Installing to create a stand-alone development environment (qesb)” • “Installing to create one or more stand-alone environments” on page 4 • “Installing WebSphere ESB, configuring deployment manager and custom profiles, and using the deployment environment wizard” on page 4 • “Installing WebSphere ESB, configuring deployment manager and custom profiles, and using the administrative console or wsadmin commands” on page 6 | You have created your chosen deployment environment and are ready to use it for your business needs. |

Installing to create a stand-alone development environment (qesb)

When installing the product software, you can choose to create the profile for a stand-alone development environment (qesb). The profile created is only suitable for use in a test scenario or to support application development.

Table 3. Installing to create a stand-alone development environment (qesb)

| Task | Where to find information | Result after completing the task |
|--|---------------------------|--|
| Install the software and select to create the profile for a stand-alone development environment (qesb) | Installing the software | The product software has been installed on your workstation, and the profile for the stand-alone development environment (qesb) has been configured. If you chose, the First Steps console is displayed for you to verify the installation, start the stand-alone server, or display the information center. |

Installing to create one or more stand-alone environments

You can install the product software then later configure stand-alone profiles by using either the Profile Management Tool or manageprofiles command-line utility.

Table 4. Installing to create one or more stand-alone environments

| Task | Where to find information | Result after completing the task |
|---|---|--|
| Install the software on each workstation that is to host a stand-alone environment, but do not select to create the profile for a stand-alone development environment (qesb). | Installing the software | The product software has been installed on your workstation. If you chose, the Profile Management Tool is displayed for you to create a profile for each stand-alone environment. |
| Create one or more stand-alone profiles. | Depending on how you want to create your profiles, see one of the following topics. <ul style="list-style-type: none"> Creating a stand-alone profile using the Profile Management Tool Creating a stand-alone profile using the manageprofiles utility | <p>You have created a stand-alone profile. This profile defines your stand-alone environment and contains command files, configuration files, and log files for that environment.</p> <p>The process that created the profile also configured the common and component-specific databases and generated the database tables required to support the stand-alone environment.</p> |

Installing WebSphere ESB, configuring deployment manager and custom profiles, and using the deployment environment wizard

You can create a typical network deployment environment by installing the product software then later configuring profiles for a deployment manager and one or more custom (managed) nodes. After profile creation you can use the deployment environment wizard to generate a pattern-based deployment environment.

Table 5. Installing and configuring WebSphere ESB using the deployment environment wizard

| Task | Where to find information | Result after completing the task |
|---|---|--|
| Install the software on each workstation that is to be used in the network deployment environment, but do not select to create the profile for a stand-alone development environment (qesb). | Installing the software | The product software has been installed on each workstation. If you chose, the Profile Management Tool is displayed for you to create profiles. |
| Design the database configuration that applies to the environment you are creating. | Creating database design files by using the database design tool Generate the design document then run the SQL scripts | The database configuration, including all of the required database tables generated by the SQL scripts exist on your system. You reference the design document from the Profile Management Tool or manageprofiles utility. You can now begin the profile creation process. For a network deployment environment configuration, you need to use the Profile Management Tool or manageprofiles to create the deployment manager profile and one or more custom (managed node) profiles. |
| Create a deployment manager profile. This task assumes that you select parameters in the Profile Management Tool or manageprofiles command-line to point to the database design document <i>that you have already created</i> . Note: If you did not create a database design document before creating a profile, you can configure the database and run the associated SQL as part of the profile creation process. | Creating the deployment manager profile | You have a deployment manager profile. |
| Start the deployment manager to verify that the start operation is successful. | For information about how to start the deployment manager, see Starting deployment managers | The deployment manager server is started. |

Table 5. Installing and configuring WebSphere ESB using the deployment environment wizard (continued)

| Task | Where to find information | Result after completing the task |
|---|---|---|
| <p>Create the custom (managed node) profiles.</p> <p>This task assumes that you are not federating nodes (making them part of the deployment manager cell) during the profile creation process. It assumes that you set parameters in the Profile Management Tool or manageprofiles command to create the custom (managed node) profiles then federate them later.</p> <p>Repeat this task for each managed node.</p> | <p>Creating custom profiles (managed nodes) using the Profile Management Tool</p> | <p>You have your custom (managed node) profiles. These nodes are to be managed by the deployment manager.</p> <p>You can now federate the node into the deployment manager cell. The managed node contains a node agent and can contain managed servers and clusters.</p> |
| <p>Federate each custom (managed) node to the deployment manager.</p> | <p>Federating custom nodes to a deployment manager</p> | <p>The custom profile is federated into the deployment manager.</p> |
| <p>Create the network deployment configuration, by using the deployment environment wizard to create a selected standard topology pattern.</p> | <p>Creating a deployment environment using a pattern</p> | <p>As part of this task you addressed any deferred configuration items and then generated the deployment environment.</p> <p>You have created your network deployment environment with a standard topology pattern.</p> |

Installing WebSphere ESB, configuring deployment manager and custom profiles, and using the administrative console or wsadmin commands

The topology patterns packaged with the software and implemented using the deployment environment wizard are intended to address a broad spectrum of business processing requirements. However, if you have a scenario that the topology patterns do not address sufficiently, you can use the administrative console to create a customized network deployment environment.

Note: This scenario is intended for users who have an advanced understanding of how to configure product components and functionality using the administrative console. Before you embark on the installation and configuration scenario described in this section, consider using the deployment environment wizard to create your network deployment environment.

Command assistance is available for a subset of administrative console actions. When available, command assistance displays the wsadmin scripting command for the last console action you performed. You can then use this data to create

wsadmin scripts that automate certain administrative tasks. For more information on command assistance, see Administrative console actions with command assistance.

Table 6. Installing and configuring WebSphere ESB using the administrative console

| Task | Where to find information | Result after completing the task |
|---|--|---|
| <p>Install the software and configure the required profiles.</p> <p>Complete the tasks listed in Table 5 on page 5 up to the task <i>Create the network deployment configuration, by using the deployment environment wizard to create a selected standard topology pattern..</i> Instead of using the deployment environment wizard, you next create a customized deployment environment using the administrative console.</p> | Table 5 on page 5 | <p>The product software has been installed on each workstation, and you have created the deployment manager and custom managed nodes and have federated the nodes into the deployment manager cell.</p> <p>You can now use the administrative console to create servers, server clusters and the components that comprise your intended network deployment environment.</p> |
| Create and configure servers and clusters, by using the administrative console. | Creating and configuring servers and clusters using the administrative console | You have created the servers and server clusters for your environment. |
| Configuring SCA support for a server or cluster. | Configuring SCA support for a server or cluster | You have configured SCA support for the server or cluster. |
| Configure Business Space. | Configuring Business Space | You have configured the Business Space component. |
| Set up the Messaging Service. | Setting up the messaging server environment | You have set up the messaging server environment. |
| Configure the JNDILookup web service. | Configuring the JNDILookup web service | You have configured the JNDILookup web service. |
| Configure Common Event Infrastructure. | Configuring Common Event Infrastructure | You have configured the Common Event Infrastructure. |
| Configure WebSphere ESB widgets for WebSphere Portal. | Configuring WebSphere ESB widgets for WebSphere Portal | You have configured widgets for WebSphere Portal. |
| Configure WebSphere Business Integration Adapters. | Configuring WebSphere Business Integration Adapters | You have configured WebSphere Business Integration Adapters. |
| You have created your network deployment environment with a custom topology. | | |

Planning for WebSphere ESB

To ensure that the system that you implement meets your needs, plan your WebSphere ESB before you introduce its software into your enterprise information system.

Assessing your requirements

To minimize rework and outages, take the time to study your current environment before you make installation and configuration decisions. Consider your current business requirements and design, the hardware and software already installed, and your current strengths and shortcomings. This planning could also help you minimize your financial investment.

Several factors determine your software needs. These factors can be organized into the following categories.

- Product hardware and software requirements, your own system resource constraints, and the availability of resources to administer and maintain your system
- Applications to be deployed to the runtime environment, and the intended use of the configured environment
- Products, and the versions of these products, to install to meet your requirements

To make wise choices for all these factors, you must understand the following concepts:

- The terminology as it applies to environment configuration
- The administrative architecture of the product that you will install, configure, administer, and maintain
- The available configuration options (through supplied patterns) and how to determine if a pattern addresses your intended use of the product
- The supported methods of implementation, including an understanding of the different task flows for installing the product and configuring the environment

You can use the information in this section to assess and analyze your current and future requirements to develop an environment to meet those requirements.

Important: For the latest information about platform-specific disk space requirements, supported operating systems, and supported database versions, click one of the following links. You can also find operating system fixes and patches that you must install to have a compliant operating system.

Resource considerations

Identify your assets to make the best use of your software and hardware resources and to make informed implementation decisions. Assess your current enterprise information system to determine whether you require any additional hardware or software to meet your business needs.

Consider the following factors:

- Familiarize yourself with current hardware and software. Prepare a list of the available assets.
- Determine the number of physical computer systems that you will use and itemize each piece of physical hardware. Record the following information:
 - Amount of installed memory

- Number and type of installed microprocessors
- External media
- Whether a particular unit can be upgraded
- Itemize the currently installed software and database applications. Record the following information:
 - Function
 - Breadth of use across the company
 - Security requirements
- Prepare a list of your current IT personnel. Determine whether you have the required expertise to install and maintain WebSphere ESB, as well as the required expertise to manage your databases. Make sure that the appropriate users have user IDs with the authorizations to successfully install all products and files.

Development and deployment version levels

When you try to determine the version levels of WebSphere ESB that you need in your environment, your decision depends on the version levels that were used when your applications were developed. Generally, applications deployed in a previous version of WebSphere ESB can run on the next available version of WebSphere ESB.

The following table describes compatibility between WebSphere ESB V7.5.1, including IBM® Integration Designer V7.5.1 (previously WebSphere Integration Developer) and IBM Process Designer V7.5.1, and prior releases.

| Task | Supported? |
|--|---|
| Deployment from WebSphere Integration Developer version 6.1.0, 6.1.2, or 6.2.0 or 7.0.0 or IBM Integration Designer 7.5 to WebSphere ESB V7.5.1. | <p>Yes.</p> <p>Important: For WebSphere Adapters V6.1.0, V6.1.2 and V6.2.0, you must install the interim fix titled <i>Mandatory adapter fix for running 6.1 and 6.2 Adapters on WPS v7.0</i>. If you do not plan to update the WebSphere Adapter to a V7.0 level, and you plan to continue to use the application with WebSphere Adapter V6.1.0, V6.1.2 or V6.2.0, you must apply this interim fix on the source environment.</p> <p>Important: Websphere Adapter for SAP V6.0.2, V6.1.0, V6.1.2 and V6.2.0 are not supported on WebSphere ESB V7.5.1. You must update Websphere Adapter for SAP to V7.0 before you can deploy any applications that use Websphere Adapter for SAP on WebSphere ESB V7.5.1. For more information specific to WebSphere Adapter for SAP, see Postmigration tasks for WebSphere ESB.</p> |

| Task | Supported? |
|--|--|
| Running WebSphere ESB V7.5.1 artifacts on WebSphere ESB 6.1.0, 6.1.2, 6.2.0, 7.0 or 7.5. | <p>No.</p> <p>Applications authored with IBM Integration Designer V7.5.1 cannot be published to or installed on WebSphere ESB 6.1.0, 6.1.2, 6.2.0, 7.0 or WebSphere ESB version 7.5 (any prior release) servers.</p> <p>Applications authored with WebSphere Integration Developer 6.1.0, 6.1.2, 6.2.0, 7.0.0 or IBM Integration Designer 7.5 and then generated in IBM Integration Designer V7.5.1 cannot be published to or installed on WebSphere ESB 6.1.0, 6.1.2, 6.2.0, 7.0 or WebSphere ESB version 7.5 servers.</p> <p>Applications generated using serviceDeploy from WebSphere ESB V7.5.1 servers cannot be installed on WebSphere ESB 6.1.0, 6.1.2, 6.2.0, 7.0 or WebSphere ESB version 7.5 servers.</p> |

Naming considerations for profiles, nodes, servers, hosts, and cells

This topic discusses reserved terms and issues you must consider when naming your profile, node, server, host, and cell (if applicable). This topic applies to distributed platforms.

Profile naming considerations

The profile name can be any unique name with the following restrictions. Do not use any of the following characters when naming your profile:

- Spaces
- Special characters that are not allowed within the name of a directory on your operating system, such as *, &, or ?.
- Slashes (/) or back slashes (\)

Double-byte characters are allowed.

Windows **Directory path considerations:** The installation directory path must be less than or equal to 60 characters. The number of characters in the *profiles_directory_path\profile_name* directory must be less than or equal to 80 characters.

Node, server, host, and cell naming considerations

Reserved names: Avoid using reserved names as field values. The use of reserved names can cause unpredictable results. The following words are reserved:

- cells
- nodes
- servers
- clusters
- applications
- deployments

Descriptions of fields on the Node and Hosts Names and Node, Host, and Cell Names pages: Table 7 describes the fields found on the Node and Host Names and Node, Host, and Cell Names pages of the Profile Management Tool, including the field names, default values, and constraints. Use this information as a guide when you are creating profiles.

Table 7. Naming guidelines for nodes, servers, hosts, and cells

| Field name | Default value | Constraints | Description |
|------------------------------------|--|--|---|
| Stand-alone server profiles | | | |
| Node name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> <i>shortHostName</i> Node <i>NodeNumber</i> where: <ul style="list-style-type: none"> <i>shortHost Name</i> is the short host name. <i>NodeNumber</i> is a sequential number starting at 01. | Avoid using the reserved names. | Select any name you want. To help organize your installation, use a unique name if you plan to create more than one server on the system. |
| Server name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> server1 | Use a unique name for the server. | The logical name for the server. |
| Host name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> The long form of the domain name server (DNS) name. | The host name must be addressable through your network. If you are planning to use Business Space, use a fully qualified host name. | Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table. |

Table 7. Naming guidelines for nodes, servers, hosts, and cells (continued)

| Field name | Default value | Constraints | Description |
|------------------------------------|--|---|---|
| Cell name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> <i>shortHostName</i> Node <i>NodeNumber</i> Cell where: <ul style="list-style-type: none"> <i>shortHost Name</i> is the short host name. <i>NodeNumber</i> is a sequential number starting at 01. | <p>Use a unique name for the cell. A cell name must be unique in any circumstance in which the product is running on the same physical workstation or cluster of workstations, such as a Sysplex. Additionally, a cell name must be unique in any circumstance in which network connectivity between entities is required either between the cells or from a client that must communicate with each of the cells. Cell names also must be unique if their name spaces are going to be federated. Otherwise, you might encounter symptoms such as a <code>javax.naming.NameNotFoundException</code> exception, in which case, you need to create uniquely named cells.</p> | All federated nodes become members of a deployment manager cell. |
| Deployment manager profiles | | | |
| Node name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> <i>shortHostName</i> Cell ManagerNode <i>Number</i> where: <ul style="list-style-type: none"> <i>shortHost Name</i> is the short host name. <i>NodeNumber</i> is a sequential number starting at 01. | <p>Use a unique name for the deployment manager. Avoid using the reserved names.</p> | The name is used for administration within the deployment manager cell. |
| Host name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> The long form of the domain name server (DNS) name. | <p>The host name must be addressable through your network. Avoid using the reserved names.</p> <p>If you are planning to use Business Space, use a fully qualified host name.</p> | Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table. |

Table 7. Naming guidelines for nodes, servers, hosts, and cells (continued)

| Field name | Default value | Constraints | Description |
|------------------------|---|--|--|
| Cell name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> <p><i>shortHostName</i> Cell <i>CellNumber</i> where:</p> <ul style="list-style-type: none"> • <i>shortHost Name</i> is the short host name. • <i>CellNumber</i> is a sequential number starting at 01. | <p>Use a unique name for the deployment manager cell. A cell name must be unique in any circumstance in which the product is running on the same physical workstation or cluster of workstations, such as a Sysplex. Additionally, a cell name must be unique in any circumstance in which network connectivity between entities is required either between the cells or from a client that must communicate with each of the cells. Cell names also must be unique if their name spaces are going to be federated. Otherwise, you might encounter symptoms such as a <code>javax.naming.NameNotFoundException</code> exception, in which case, you need to create uniquely named cells.</p> | All federated nodes become members of the deployment manager cell, which you name in the Node, Host, and Cell Names page of the Profile Management Tool. |
| Custom profiles | | | |
| Node name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> <p><i>shortHostName</i> Node <i>NodeNumber</i> where:</p> <ul style="list-style-type: none"> • <i>shortHost Name</i> is the short host name. • <i>NodeNumber</i> is a sequential number starting at 01. | <p>Avoid using the reserved names.</p> <p>Use a unique name within the deployment manager cell.</p> | The name is used for administration within the deployment manager cell to which the custom profile is added. Use a unique name within the deployment manager cell. |
| Host name | <div>Linux</div> <div>UNIX</div> <div>Windows</div> <p>The long form of the domain name server (DNS) name.</p> | <p>The host name must be addressable through your network.</p> <p>If you are planning to use Business Space, use a fully qualified host name.</p> | Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table. |

Host name considerations:

The host name is the network name for the physical workstation on which the node is installed. The host name must resolve to a physical network node on the server. When multiple network cards exist in the server, the host name or IP address must resolve to one of the network cards. Remote nodes use the host name to connect to and to communicate with this node.

WebSphere ESB is compliant to both Internet Protocol version 4 (IPv4) and version 6 (IPv6). Wherever you can enter IP addresses in the administrative console, or elsewhere, you can do so in either format. Note that if IPv6 is implemented on your system you must enter the IP address in IPv6 format, and conversely, if IPv6 is not yet available to you, enter IP addresses in IPv4 format. For more information on IPv6 see the Official IPv6 Web site.

The following guidelines can help in determining the appropriate host name for your workstation:

- Select a host name that other workstations can reach within your network.
- Do not use the generic identifier, localhost, for this value.
- Do not attempt to install WebSphere ESB products on a server with a host name that uses characters from the double-byte character set (DBCS). DBCS characters are not supported when used in the host name.
- Avoid using the underscore (_) character in server names. Internet standards dictate that domain names conform to the host name requirements described in Internet Official Protocol Standards RFC 952 and RFC 1123. Domain names must contain only letters (upper or lower case) and digits. Domain names can also contain dash characters (-) as long as the dashes are not on the ends of the name. Underscore characters (_) are not supported in the host name. If you have installed WebSphere ESB on a server with an underscore character in the server name, access the server with its IP address until you rename it.

If you define coexisting nodes on the same computer with unique IP addresses, define each IP address in a domain name server (DNS) look-up table. Configuration files for servers do not provide domain name resolution for multiple IP addresses on a workstation with a single network address.

The value that you specify for the host name is used as the value of the hostName property in configuration documents. Specify the host name value in one of the following formats:

- Fully qualified domain name servers (DNS) host name string, such as xmachine.manhattan.ibm.com
- The default short DNS host name string, such as xmachine
- Numeric IP address, such as 127.1.255.3

The fully qualified DNS host name has the advantages of being totally unambiguous and flexible. You have the flexibility of changing the actual IP address for the host system without having to change the server configuration. This value for host name is particularly useful if you plan to change the IP address frequently when using Dynamic Host Configuration Protocol (DHCP) to assign IP addresses. A disadvantage of this format is being dependent on DNS. If DNS is not available, then connectivity is compromised.

The short host name is also dynamically resolvable. A short name format has the added ability of being redefined in the local hosts file so that the system can run the server even when disconnected from the network. Define the short name to 127.0.0.1 (local loopback) in the hosts file to run disconnected. A disadvantage of

the short name format is being dependent on DNS for remote access. If DNS is not available, then connectivity is compromised.

A numeric IP address has the advantage of not requiring name resolution through DNS. A remote node can connect to the node you name with a numeric IP address without DNS being available. A disadvantage of this format is that the numeric IP address is fixed. You must change the setting of the `hostName` property in configuration documents whenever you change the workstation IP address. Therefore, do not use a numeric IP address if you use DHCP, or if you change IP addresses regularly. Another disadvantage of this format is that you cannot use the node if the host is disconnected from the network.

Preparing necessary security authorizations

Depending on your security policy, you might need a user ID and password to complete tasks such as creating files and folders and accessing the database. Prepare secure user IDs to prevent problems when the servers attempt to access protected data.

Before you begin

- Complete the design of your database.
- Determine the authentication system to use, for example, Lightweight Directory Access Protocol (LDAP).
- Determine what controls are in place that affect the authorizations required for your WebSphere ESB installation.
- Identify the systems on which you are installing the product.

About this task

The security policies for your site enables global security which indicates that you require specific authorities to install software, create databases or tables, or access databases. To successfully install and operate the product you must do these steps.

Procedure

To prepare security authorizations for an IBM Business Process Manager database, complete the following steps:

- Prepare a list of user IDs and passwords that have authority to install software on the systems. You must run the installation wizards for WebSphere ESB user IDs that have the authority to create files and folders.
- Prepare a list of user IDs, passwords, and roles that are needed for daily operations of the system:
 - Administrative console user IDs and roles to limit capabilities. You can have user IDs for configuring, administering, or monitoring roles.
 - User IDs for each system bus to authenticate system communications.
- Prepare a list of user IDs and passwords that the system uses to access the database tables that it uses during operation.
- Optional: Prepare a list of user IDs and passwords that the system uses to create databases or database tables during installation. Your site policies might restrict this authority to the database administrator. In this case, you must provide generated scripts to the administrator to create the databases or database tables.

Results

You can install and operate your servers in a secure environment.

Installation directories for the product and profiles

The installation directories for WebSphere ESB are represented by several variables. The meaning of those variables can differ for a number of factors.

Variables used in the documentation

Several variables representing specific default directories are used throughout the documentation. These file paths are default locations. You can install the product and other components and create profiles in any directory for which you have write access. Multiple installations of WebSphere ESB products or components require multiple locations.

Here are the main variables used in the documentation:

Linux **UNIX** **Windows** *install_root*
Installation location of WebSphere ESB. WebSphere ESB is always installed in the same location as the WebSphere Application Server Network Deployment installation with which it is associated.

profile_root
Location of a WebSphere ESB profile.

How variable meanings can differ

The meaning of variables used to represent installation directories can differ based on whether you are installing the product on a clean workstation or on a workstation that has an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment. The variables can also differ depending on whether you are performing the installation as a root (Administrator on a Windows system) or nonroot user.

Linux **UNIX** **Windows** Limitations of nonroot installers

Root, Administrator, and nonroot users can install the product. The default directories the installation program provides differ based on whether the user has root (Administrator) privileges. Root and Administrator users can register shared products and install into system-owned directories (globally shared resources that are available to all users), while nonroot users cannot. Nonroot users can install only into directories they own.

Default directories for Installation

The following tables show the default installation locations of the product and its profiles. If you choose to install WebSphere ESB on top of an existing supported version of WebSphere Application Server or WebSphere Application Server Network Deployment, WebSphere ESB is installed into the same location. Table 8 shows the default installation root directory in such a case for both root (Administrator) and nonroot users.

Table 8. shows the default installation root directory into which the installation program installs both WebSphere ESB and WebSphere Application Server for both root (Administrator) and nonroot users.

| Default <i>install_root</i> for root or Administrator users | Default <i>install_root</i> for nonroot users |
|---|--|
| AIX /usr/IBM/WebSphere/AppServer | AIX <i>user_home</i> /IBM/WebSphere/AppServer |

Table 8. shows the default installation root directory into which the installation program installs both WebSphere ESB and WebSphere Application Server for both root (Administrator) and nonroot users. (continued)

| Default <i>install_root</i> for root or Administrator users | Default <i>install_root</i> for nonroot users |
|---|---|
| Linux Solaris /opt/IBM/WebSphere/AppServer | Linux Solaris <i>user_home</i> /IBM/WebSphere/AppServer |
| Windows C:\Program Files\IBM\WebSphere\AppServer | Windows <i>user_home</i> \IBM\WebSphere\AppServer |

Table 9. shows the default installation directory for a profile named *profile_name* for both root (Administrator) and nonroot users.

| Default <i>profile_root</i> for root or Administrator users | Default <i>profile_root</i> for nonroot users |
|--|---|
| AIX /usr/IBM/WebSphere/AppServer/profiles/ <i>profile_name</i> | AIX <i>user_home</i> /IBM/WebSphere/AppServer/profiles/ <i>profile_name</i> |
| Linux Solaris /opt/IBM/WebSphere/AppServer/profiles/ <i>profile_name</i> | Linux Solaris <i>user_home</i> /IBM/WebSphere/AppServer/profiles/ <i>profile_name</i> |
| Windows C:\Program Files\IBM\WebSphere\AppServer\profiles\ <i>profile_name</i> | Windows <i>user_home</i> \IBM\WebSphere\AppServer\profiles\ <i>profile_name</i> |

Default installation directories for Installation Manager

Table 10 shows two default directories related to the Installation Manager tool.

The directories under **Installation directory** are the defaults (per platform) into which the launchpad application installs Installation Manager.

The directories under **Agent data location directory** are the defaults (per platform) used by Installation Manager for data associated with the application, such as the state and history of operations performed by Installation Manager.

Values are given for both root (Administrator) and nonroot users.

For more information about the Agent data location, see Agent data location in the Installation Manager documentation. For more information on other defaults for Installation Manager, see Installing as an administrator or non-administrator in the Installation Manager documentation.

Table 10. Installation Manager default installation directories

| Defaults for root or Administrator users | Defaults for nonroot users |
|---|--|
| Installation directory: | Installation directory: |
| Linux /opt/IBM/InstallationManager/eclipse | Linux <i>user_home</i> /IBM/InstallationManager/eclipse |
| UNIX /opt/IBM/InstallationManager/eclipse | UNIX <i>user_home</i> /IBM/InstallationManager/eclipse |
| Windows C:\Program Files\IBM\Installation Manager\eclipse | Windows C:\Documents and Settings\ <i>userID</i> \IBM\Installation Manager\eclipse |
| | Vista Windows 7 C:\ProgramData\IBM\Installation Manager |
| Agent data location directory: | Agent data location directory: |
| Linux /var/ibm/InstallationManager | Linux <i>user_home</i> /var/ibm/InstallationManager |

Table 10. Installation Manager default installation directories (continued)

| Defaults for root or Administrator users | Defaults for nonroot users |
|--|--|
| UNIX /var/ibm/InstallationManager | UNIX <i>user_home</i> /var/ibm/InstallationManager |
| Windows C:\Documents and Settings\All Users\Application Data\IBM\Installation Manager | Windows C:\Documents and Settings\ <i>userID</i> \Application Data\IBM\Installation Manager |
| Vista Windows 7 C:\ProgramData\IBM\Installation Manager | Vista Windows 7 C:\Users\ <i>userID</i> \AppData\Roaming\IBM\Installation Manager |

Choosing a stand-alone or network deployment environment

Choose a stand-alone environment to evaluate the product or to support development of applications and services. Choose a network deployment environment when your production environment needs additional features such as capacity, availability, scalability, and failover support.

A stand-alone environment is the easiest to install and configure, and requires little planning. A network deployment environment needs more extensive installation and configuration tasks that can involve several roles.

For a network deployment environment, you should carefully plan the characteristics with a goal of meeting the requirements of the work that business applications and services are to perform on it. There are multiple aspects to consider, including the following:

- Number of physical workstations and hardware resources that you require
- Number of clusters and cluster members required to support your business
- Number of databases required
- Authentication roles and security considerations
- The method that you will use to implement the deployment environment
- Other supporting resources such as a user registry (for security), one or more HTTP servers (for web content), necessary firewalls, load balancers, and so on.

Stand-alone environment

A stand-alone environment contains a server to run your service requester and provider enterprise applications and the mediation modules that they use. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

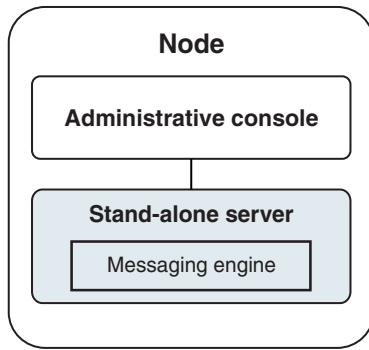


Figure 2. A stand-alone environment

To evaluate the product or to support development of applications and services, you can install samples to deploy a sample solution to the stand-alone server. You can explore the resources used for this sample in the administrative console.

To start with a stand-alone environment and then to include it into a network deployment environment, federate it into a deployment manager cell. You can do so only if no other nodes have been federated to that cell.

When you install the product software, you can choose to create the profile for a stand-alone development environment (qesb). The profile that is created is suitable only in a test scenario or to support application development. For a scenario in which you want a stand-alone server environment for production purposes, install the product software. Then use the Profile Management Tool or **manageprofiles** command-line utility to configure the stand-alone profiles.

Network deployment environment

A network deployment environment contains a collection of interconnected servers and clusters to run your service requester and provider enterprise applications and their mediation modules. The environment can also include application servers on WebSphere Application Server.

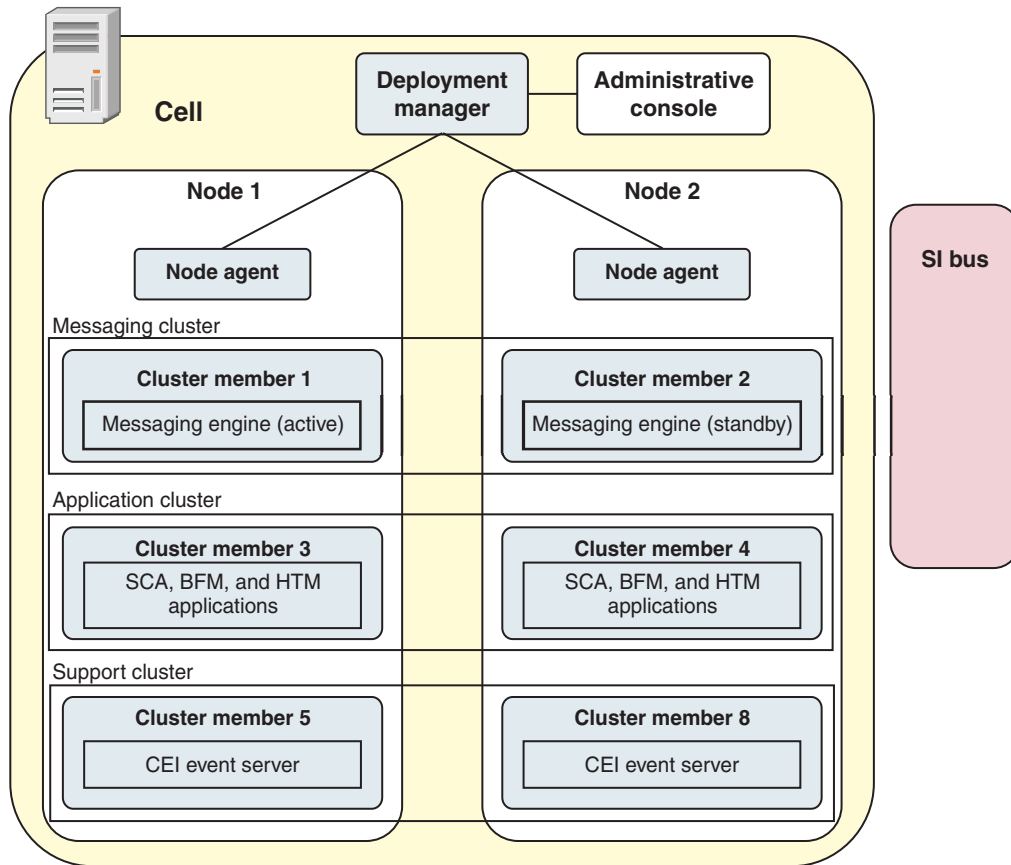


Figure 3. A network deployment environment

The servers and clusters run on one or more managed *nodes*, each of which corresponds to a logical or physical computer system.

Servers can be grouped into *clusters* to support load-balancing and failover.

A deployment environment of interconnected servers or clusters provides performance, availability, scalability, isolation, security, and stability characteristics that cannot be provided by a stand-alone server. In addition, you can manage all the servers or clusters from a centralized *deployment manager*.

A complete collection of servers and clusters managed by a deployment manager is configured and managed as a *deployment environment*.

To install a network deployment environment, install the product software, and then configure profiles for a deployment manager and one or more custom (managed) nodes. Later, you can create the deployment environment to be managed. You can create a *standardized* deployment environment from provided topology patterns, or you can configure clusters and servers to create a *customized* deployment environment.

How intended usage affects your choice of stand-alone or network deployment cluster topology pattern

The following table shows how the intended use of WebSphere ESB affects your choice of stand-alone or a network deployment cluster topology pattern, and the associated amount of planning involved:

Table 11. Choice of stand-alone or network deployment cluster topology pattern for intended use of WebSphere ESB

| Intended use | Configuration path and planning activities |
|---|--|
| A single server Unit Test Environment (UTE) | The stand-alone profile configuration path, with little planning required. |
| A clustered test environment | Standard Remote Messaging and Remote Support topology pattern of network deployment environment, with little planning required. |
| A production environment, with good flexibility | Standard Remote Messaging and Remote Support topology pattern of network deployment environment, with little planning required. |
| A highly optimized production environment | A customized topology that addresses unique processing requirements and business requirements. Detailed planning required as described in this section of the documentation. |

For more information about selecting an appropriate cluster topology pattern, refer to the related concepts links.

Planning your network deployment environment

Setting up a network deployment environment involves many decisions, such as the number of physical workstations and the type of pattern you choose. Each decision affects how you set up your deployment environment.

Before you begin

Before you plan your deployment environment complete the following tasks:

- Choose a database type
- Identify available resources
- Identify necessary security authorizations

About this task

When you plan the layout of interconnected servers, you must make some decisions. These decisions influence trade-offs that you make between the available hardware and physical connections, the complexity of the management and configuration and requirements such as performance, availability, scalability, isolation, security, and stability.

Procedure

1. Identify the functional requirements of the deployment environment.
 - a. Identify the features or runtime capabilities of your deployment environment.
 - b. Identify the component types that you will deploy.

Consider the component types and the interactions between components as part of the requirements.
 - c. Identify the import and export implementation types and transports.

Consider the resources needed for the databases or Java Message Service (JMS) resources and the need for business events and their transmission mechanism.

- d. Identify any functional requirements that are not related to applications.

Consider security servers, routers, and any other hardware or software requirements to handle business events.

2. Identify the capacity and performance requirements for your environment.
3. Decide on the number of physical servers that you need for each function.
4. Design your deployment environment.

Decide on the pattern. For WebSphere ESB, you can select one of four established topology patterns:

- Single Cluster
- Remote Messaging
- Remote Messaging and Remote Support
- Remote Messaging, Remote Support, and Web

If none of these patterns meets your needs, you can use the administrative console to create a custom deployment environment.

For more information about the patterns and the differences between them, see “Topologies of a network deployment environment” on page 25.

5. Understand the methods available to you for configuring your deployment environment.

You can configure the following types of deployment environments for WebSphere ESB:

- A standardized network deployment environment

A standardized network deployment environment is based on a topology pattern template included with the software and implemented by using the Deployment Environment configuration wizard or wsadmin commands.

You can use the Deployment Environment Configuration wizard to create clusters with the Single Cluster, Remote Messaging, Remote Messaging and Remote Support, and (if applicable) Remote Messaging, Remote Support, and Web cluster topology patterns.

- A customized network deployment environment

A customized network deployment environment is a configuration that you create from the administrative console, as opposed to a “template-based” configuration from the Deployment Environment wizard.

You should create a customized network deployment environment only if the topology patterns that are included with the software do not meet your configuration needs.

As is the case with the standardized environment, you can create a customized network deployment environment with wsadmin.

Overview: Deployment environment topologies and patterns

A network deployment environment can have many topologies, and can be created from several standard topology patterns.

What is a topology?

A topology is the physical layout of the *deployment environment* required to meet your business needs for capacity, availability, and scalability.

Many factors affect how you design and implement your topology. For example, you must consider business and application requirements, resource requirements and constraints, the intended purpose of the environment, and the operating system.

WebSphere ESB includes patterns for the following topologies, which you can use to address many business scenarios, from proof-of-concept (POC) to a fully functional production environment:

- Single Cluster
- Remote Messaging
- Remote Messaging and Remote Support
- Remote Messaging, Remote Support, and Web

Each topology pattern has certain design characteristics that address a particular business need. For example, on distributed systems, the Single Cluster topology pattern is typically used for a testing or proof of concept scenario. On z/OS® systems, this topology pattern is the default pattern and can be used in production environments.

The design characteristics of each topology have been captured as *topology patterns* that are supplied as configuration templates with the product.

You are not obligated to use a standardized (IBM-supplied) topology pattern. If none of the topology patterns address your specific need, you can create a custom topology pattern.

The purpose of deployment environment patterns

A deployment environment topology pattern specifies the constraints and requirements of the components and resources involved in a deployment environment. There are IBM-supplied topology pattern for each topology layout. These topology patterns provide rules and guidelines for component interaction that are characteristic of the most commonly used BPM topology patterns. The IBM-supplied topology patterns are based on well-known and tested configuration scenarios. They contain a repeatable and automated method of creating a deployment environment. Each topology pattern is designed to meet the configuration requirements and business needs of the associated topology. Using topology patterns helps you create a deployment environment in the most straightforward way.

Because the deployment environment topology patterns represent recommended topologies with component configurations that work together, you can be sure that you are building a fully functional deployment environment. You can use the configuration rules of a deployment environment topology pattern to generate a fast path configuration. This action is possible because many design decisions are implemented in the topology pattern; for example, which components to configure, and which default parameters and resources are needed.

Each supplied deployment environment topology pattern addresses a specific set of requirements. Most requirement sets can be met when you use one of these topology patterns. To select a topology pattern, complete all of the following steps:

- Understand the requirements of the business solution that you are creating.
- Review and understand the capabilities and characteristics of the IBM-supplied topology patterns.

- Decide which topology pattern to use.
If none of the WebSphere ESB topology patterns suit your needs, you can use the administrative console or scripting (wsadmin commands) to create a customized topology pattern.

Databases and deployment environments

Before you can create and configure a network deployment environment, you must configure your database and create the required database tables. At a minimum, to use WebSphere ESB, you need to configure the common database.

For a stand-alone server configuration, the installation option configures these databases and creates the required database tables automatically.

For a network deployment environment (customized deployment environment or standardized deployment environment), you or your database administrator must configure the databases outside the installer. Additional databases are required to support additional functionality. For example, if your WebSphere ESB configuration includes Business Space or Common Base Event monitoring, you or your database administrator must configure these databases and use the supplied utilities or scripts to create the required database tables. You must do this configuration before you can create the network deployment environment.

For more information, see *Planning your database configuration*.

Functions of IBM-supplied deployment environment topology patterns

Any WebSphere ESB deployment contains a basic set of functions that together form a complete production environment.

To design a robust deployment environment, you must understand the functionality that each cluster can provide in an IBM-supplied topology pattern or custom deployment environment. You can allocate a specific type of function (for example, the support infrastructure function) to a particular cluster. Understanding the functions can help you choose the deployment environment topology pattern that best meets your needs.

For network deployment, clusters can collaborate to provide specific functionality to the environment. Depending on your requirements, you assign specific functions to each cluster within the deployment environment, to provide performance, failover, and capacity.

The clusters configured in a deployment environment provide the following functions.

The functions can exist in a single cluster, or can be spread across multiple clusters. Each standardized (IBM-supplied) topology pattern creates a different number of clusters to support the functions. The number of clusters in your deployment environment depends on the topology pattern that you are using.

Application deployment target

An application deployment target is the set of servers (cluster) to which you install your applications (for example, human tasks, business processes, and mediations). Depending on which deployment environment topology pattern you choose, the application deployment target might also provide messaging infrastructure and supporting infrastructure functions.

In a Single Cluster topology pattern, the application deployment target provides the entire functionality of the deployment environment.

Supporting infrastructure

The supporting infrastructure includes the Common Event Infrastructure (CEI) server and other infrastructure services used to support your environment and manage your system.

Important: You must use a custom profile with the same product functionality for this node as you did for the application deployment target cluster.

Messaging engine infrastructure

The messaging infrastructure is the set of servers (cluster) where the messaging engines are located. The messaging infrastructure is used to provide asynchronous messaging support for your applications and for the internal messaging needs of the WebSphere ESB components. The messaging engines enable communication among the nodes in the deployment environment. Your cluster can consist of members on nodes created with WebSphere Application Server instead of WebSphere ESB if the cluster solely provides the messaging function.

Web application infrastructure

Consists of a cluster where the web-based component Business Space is located.

Topologies of a network deployment environment

A topology is the physical layout of the deployment environment. You can create the topology that best addresses your business needs by choosing one of the patterns provided by IBM or by creating your own customized pattern.

Single Cluster topology pattern:

The *Single Cluster* topology pattern is an IBM-supplied topology pattern. In a Single Cluster topology pattern, all the functions of the deployment environment are combined into a single cluster.

This is the default pattern for WebSphere ESB for z/OS.

A Single Cluster topology pattern is ideal for limited hardware. Because all the components are installed in the same cluster, fewer physical machines are required. However, because each server instance must run the supporting applications and your integration applications, you need more memory for the individual Java Virtual Machines (JVMs). In addition, one or more members of the cluster must also run the messaging engines required for asynchronous interactions. Thus, the Single Cluster topology pattern is typically used for proof of concept, development, and testing environments.

Combining all aspects of the WebSphere ESB environment into a single cluster has other implications aside from the increased memory requirements.

- Because asynchronous interactions (involving JMS and MQ/JMS bindings) can make extensive use of the messaging infrastructure, a single cluster environment is not ideal for applications with these components.
- Any messaging requirements must be kept to a minimum with this topology pattern (except for z/OS).

- Service Component Architecture (SCA) internal asynchronous invocations, the Java Message Service (JMS), and MQ messaging bindings do not support multiple messaging engines in the same cluster.

If necessary, choose one of the other topology patterns in which the messaging infrastructure is in a separate cluster from the application deployment target.

The Single Cluster topology pattern is suitable for scenarios that are focused on running applications and on synchronous invocations. This topology pattern is also not ideal if you intend to make extensive use of the Common Event Infrastructure (CEI). Generating events and CEI-related messaging traffic, places an additional burden on the cluster members.

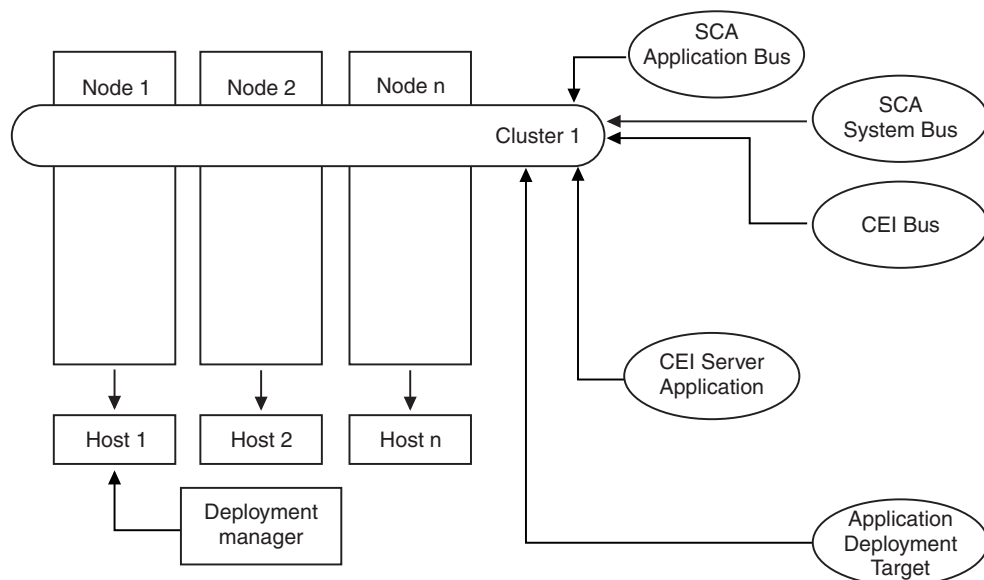
From an administrative and scalability perspective, the Single Cluster topology pattern has advantages. A single cluster where each member runs all the WebSphere ESB components are simpler to administer. Instead of several server instances in multiple clusters, you have a single cluster with fewer members. If the needs of your environment grow, scaling the infrastructure is a simple matter of adding additional nodes and cluster members. Thus, the process of adding capability is simple, but all components are scaled at the same rate. For example, each additional cluster member adds CEI processing whether you need it or not. If the messaging engines spread across server members use policies, there could be some additional administrative effort in creating and maintaining the policies.

In a Single Cluster topology pattern, all deployment environment functions and components run on a single cluster:

- Service Component Architecture (SCA) application bus members
- SCA system bus members
- CEI bus members
- CEI server
- Application deployment target

You configure the application deployment target to support SCA applications.

See the following graphical representation of the Single Cluster topology pattern.



Remote Messaging topology pattern:

The *Remote Messaging* topology pattern is an IBM-supplied topology pattern. In a Remote Messaging topology pattern, the deployment environment functions are divided between two separate clusters.

The Remote Messaging topology pattern provides a separate cluster for the messaging function. This topology pattern is suitable for scenarios involving asynchronous invocations, because the cluster can be scaled for this load. The components are divided between the two clusters.

For environments that must support numerous asynchronous interactions, a Remote Messaging topology pattern has advantages over the Single Cluster topology pattern.

Separating the messaging infrastructure into a separate cluster removes the messaging overhead from the application target cluster. When you have a separate messaging infrastructure, you need less memory for the application target cluster members. This topology pattern also differs from the Single Cluster topology pattern in terms of the hardware required. Because there are two clusters with multiple cluster members, the hardware requirements are greater for distributed environments.

From an administrative perspective, the requirements for the Remote Messaging topology pattern are greater than the requirements for the Single Cluster topology pattern. Additional clusters and additional cluster members increase the administrative effort required. In addition, because you are distributing the messaging engines across the members of the messaging cluster, you must create and maintain policies.

In the Remote Messaging topology pattern, the supporting applications and the Common Event Interface (CEI) components are still part of the application target cluster. Thus, for environments that make extensive use of CEI, the Remote Messaging topology pattern might not be ideal either. For small to medium-sized businesses, or for businesses without extensive monitoring or auditing requirements, this topology pattern is generally suitable.

The scalability options for the Remote Messaging topology pattern are as straightforward as the options for the Single Cluster topology pattern. Because the messaging engines are subject to one of *n* policies (each messaging engine is active on only one server), adding additional members to the messaging cluster has little effect. When you use policies to spread the messaging engines across server members, you can divide the messaging burden across a maximum of three servers. (The SCA.SYSTEM and SCA.APPLICATION engines are active on the same server.) Thus, adding more than three cluster members to the messaging cluster has no effect on the processing capability of the messaging infrastructure. Scaling the application target cluster is relatively easy. If you need additional processing capability for your applications or for the supporting infrastructure, you can add additional nodes and members to the application target cluster.

Messaging infrastructure cluster:

- Service Component Architecture (SCA) application bus members
- SCA system bus members
- CEI bus members

Application deployment target cluster:

- CEI server application
- Application deployment target

You configure the application deployment target to support SCA applications.

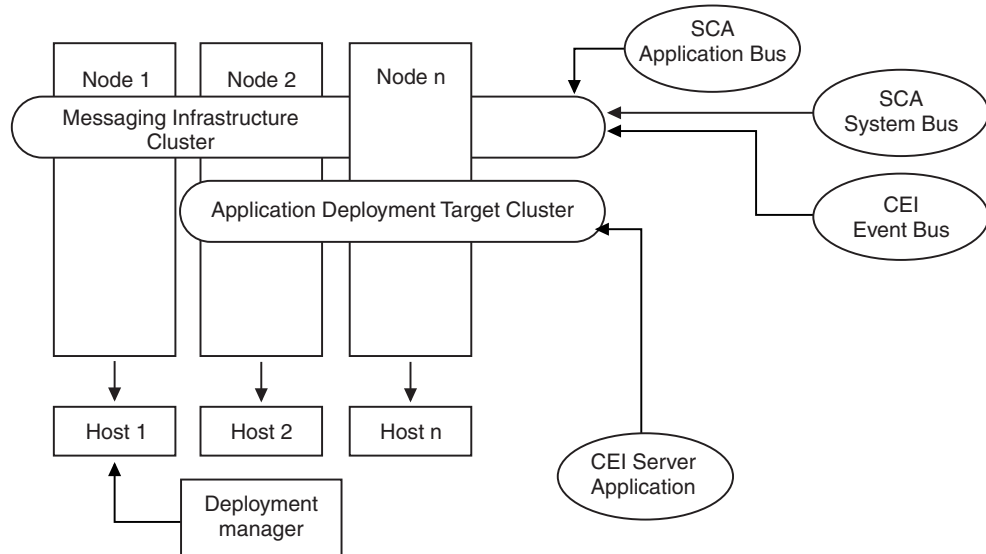


Figure 4. Remote Messaging topology pattern

Remote Messaging and Remote Support topology pattern:

The *Remote Messaging and Remote Support* topology pattern is an IBM-supplied topology pattern. In a Remote Messaging and Remote Support topology pattern, the deployment environment functions are divided among three separate clusters.

With this three-cluster topology pattern, resources are allocated to the cluster that handles the highest loads. This topology pattern is the most flexible and versatile, and is preferred by most users (except for z/OS). The components are divided among the three clusters.

For many customers with large computing infrastructures, the Remote Messaging and Remote Support topology pattern is the preferred environment. The hardware requirements for distributed platforms are more intensive. However, you have greater flexibility in adjusting and tuning memory usage for the Java virtual machines (JVMs) when you have three or more clusters with multiple members performing specific functions.

When you create three clusters, each with its own functions and applications, you add an additional administrative burden. As you add clusters and cluster members, your performance tuning plan and the troubleshooting burden can expand greatly. Spreading messaging engines across the members of the messaging cluster also adds to the administrative burden associated with creating and maintaining policies.

From a scalability standpoint, the Remote Messaging and Remote Support topology pattern provides the most flexibility. Because each of the distinct functions within WebSphere ESB is divided among the three clusters, you can

pinpoint performance bottlenecks and adjust the cluster size fairly easily. If you need additional Common Event Interface (CEI) processing, you can simply add a node and cluster member to the support cluster. Because expanding the messaging infrastructure beyond three cluster members has no effect on processing capability, the scalability limitations of the Remote Messaging topology pattern also apply to the Remote Messaging and Remote Support topology pattern.

As with the Remote Messaging topology pattern, the Remote Messaging and Remote Support topology pattern provides an ideal environment for asynchronous interactions (including JMS and MQ/JMS bindings).

Because the application target cluster runs your business integration applications only, performance tuning and diagnostics are much simpler than in the topology patterns where the application target cluster has additional responsibilities. The Remote Messaging and Remote Support topology pattern is also ideal for environments that make extensive use of CEI for monitoring and auditing (including environments with IBM Business Monitor). When you separate the support infrastructure into its own cluster, you have a dedicated set of cluster members for CEI and for supporting applications.

Messaging infrastructure cluster:

- Service Component Architecture (SCA) application bus members
- SCA system bus members
- CEI bus members

Supporting infrastructure cluster:

- CEI server application

Application deployment target cluster:

- Application deployment target

You configure the application deployment target to support SCA applications.

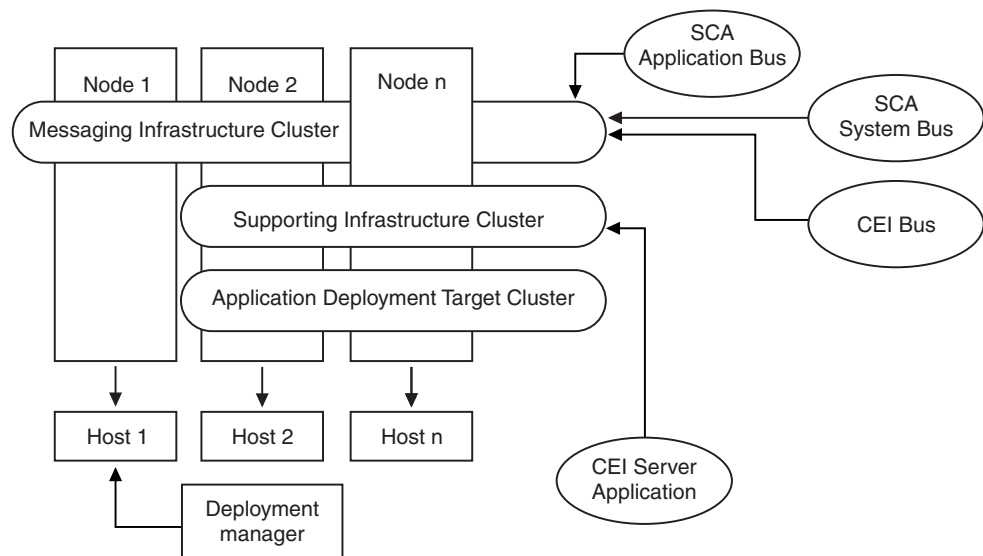


Figure 5. Remote Messaging and Remote Support topology pattern

Resource allocation example

The following figure shows one way to use the Remote Messaging and Remote Support topology pattern to allocate resources. The figure shows three hosts. Host A has Server 1 and Server 3; Host B has Server 2, Server 4, and Server 5 and Host C has Server 6 and Server 7. Because the heaviest load for this installation is for application use, more resources for Server 1, Server 2, and Server 6 are allocated for the application deployment target cluster (Cluster 3) than for the other functions.

Important: Load balancing is not available for the default configuration Remote Messaging and Remote Support topology pattern. That configuration uses a single messaging engine bus, while the load balancing feature requires at least two messaging engine buses.

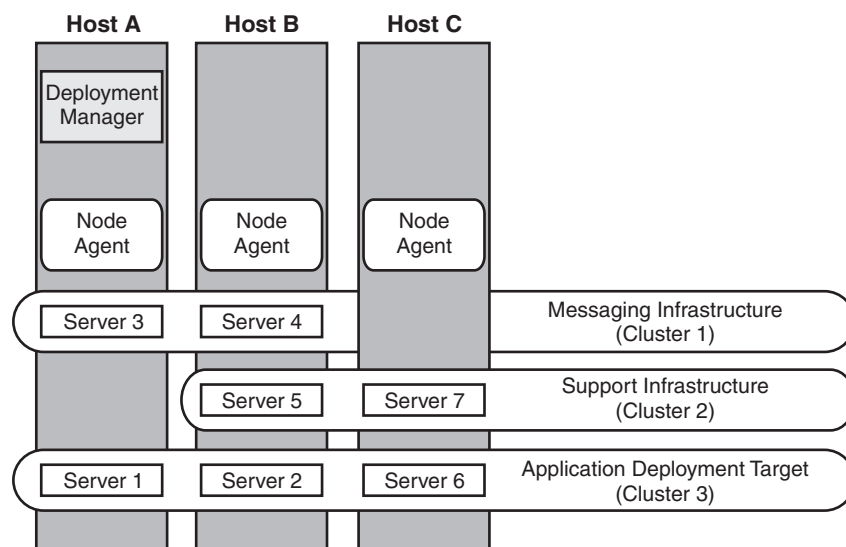


Figure 6. Resource allocation example

Remote Messaging, Remote Support, and Web topology pattern:

The *Remote Messaging, Support and Web* topology pattern is an IBM-supplied topology pattern. In a Remote Messaging, Support and Web topology pattern, the deployment environment functions are divided among four separate clusters.

This four-cluster topology pattern is similar to the Remote Messaging and Remote Support topology pattern, except that supporting web applications reside on their own cluster.

Application deployment cluster:

- Application deployment target
You configure the application deployment target to support Service Component Architecture (SCA) applications.

Messaging infrastructure cluster:

- SCA application bus members
- SCA system bus members

- Common Event Interface (CEI) bus members

Supporting infrastructure cluster:

- CEI server application

Web application cluster:

- Business Space
- REST API Services

In a Remote Messaging, Support and Web topology pattern, the deployment environment functions are divided among four separate clusters. One cluster is used for messaging functionality, one cluster for support functionality, one cluster for applications, and one cluster for web-based functions.

In addition to the ability to precisely control the individual components in your environment, the advantages of this topology pattern are similar to the advantages of the Remote Messaging and Remote Support topology pattern.

See the following graphical representation of a Remote Messaging, Support and Web topology.

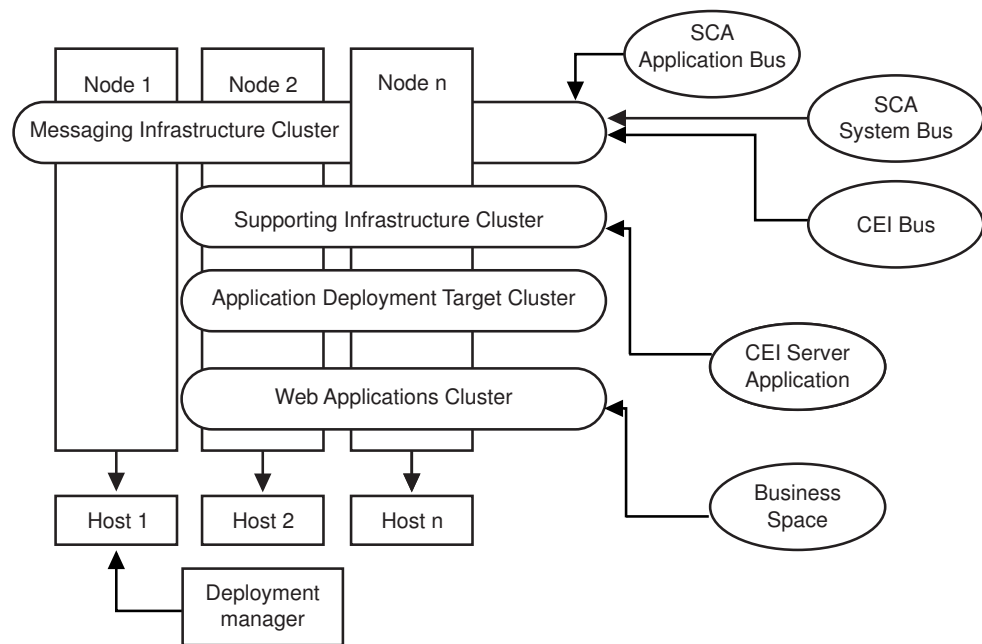


Figure 7. Remote Messaging, Support and Web pattern

Customized topology:

A customized topology addresses the processing and business requirements unique to your situation. It is not an IBM-supplied topology pattern, but rather a topology that you create and then tailor to your specific needs.

If you need to define your own deployment environment topology, a customized topology is by far the most flexible. The IBM-supplied topologies (Single Cluster, Remote Messaging, Remote Messaging and Remote Support, and Remote Messaging, Remote Support, and Web), deploy all WebSphere ESB components to their default locations. You might or might not need the additional overhead

associated with these components. For example, if your organization does not need Common Event Interface (CEI), you could create a custom topology that removes CEI support from your environment.

Except for the ability to control the components deployed in your environment, the advantages of custom topologies are similar to the advantages of the Remote Messaging and Remote Support topology. The disadvantages are also similar.

Important: Creating a customized network deployment environment is more labor-intensive than using an IBM-supplied topology pattern, which can be created from the Deployment Environment configuration wizard. Before you attempt to create a customized network deployment environment, make sure that none of the IBM-supplied topologies address your needs. You should attempt to create a customized network deployment environment only if you have a solid understanding of the features and functions of the administrative console.

Considerations for selecting a topology

Selecting an appropriate topology for your deployment environment depends upon several factors.

When you select a topology pattern, consider the following factors:

- Available hardware resources
- Application invocation patterns
- How heavily you intend to use the Common Event Infrastructure (CEI)
- Individual scalability requirements
- Administrative effort involved

In general, the Remote Messaging and Remote Support topology pattern is the most suitable production topology pattern, but the choice ultimately depends upon your individual requirements.

As you plan for your production environment, consider carefully the advantages and disadvantages of each of the common topology patterns.

Condensed topology pattern selection criteria

Consider the information listed in the following table, which is a quick guide to selecting your production topology. This table provides a condensed list of the advantages and disadvantages of each of the topology patterns.

For information about which BPM products support the supplied topology patterns, see *Topology patterns and supported BPM product features*.

Table 12. Considerations for selecting a topology for your deployment environment

| Consideration | Topology Pattern | | | |
|---------------------------------------|--|---|---|---|
| | Single cluster | Remote Messaging | Remote Messaging and Remote Support | Remote Messaging, Remote Support and Web |
| Number of clusters to maintain | One cluster for all components | One cluster for applications and for the support infrastructure One cluster for messaging | One cluster for applications One cluster for the support infrastructure One cluster for the support infrastructure | One cluster for applications One cluster for Web interfaces One cluster for support infrastructure One cluster for messaging |
| Hardware requirements | Can be implemented on limited hardware | More hardware required for distributed environments | More hardware required for distributed environments | Most hardware intensive |
| Asynchronous interactions | Use should be minimal | Use must be balanced against resource availability | Ideal environment for asynchronous interactions | Ideal environment for asynchronous interactions |
| Heavy CEI activity | Not recommended (Light CEI use should be balanced against resource usage.) | Not recommended (Light CEI use should be balanced against resource usage.) | Ideal environment for heavy CEI use | Ideal environment for heavy CEI use |
| Administrative burden | Relatively small | Requires additional effort | Requires additional administrative effort | Requires most administrative effort |
| Scalability | All components scaled at the same rate | Messaging cluster scalability limited (no benefit beyond three servers) All other components scaled at the same rate | Easy to scale All functions separated Messaging cluster scalability still limited (no benefit beyond three servers) | Easiest to scale All functions separated Messaging cluster scalability still limited (benefit comes when other BPM products are introduced) |

Topology patterns and supported product features

A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

If you are using the Deployment Environment Configuration wizard on the administrative console to create the deployment environment, the availability of topology patterns on which you base your deployment environment varies depending on the following conditions and configuration decisions:

- The platform on which you have installed WebSphere ESB
- The primary deployment environment feature and the complimentary feature

Table 13 on page 34 shows the relationship between the topology patterns and product features.

Table 13. Available supplied patterns and their relationship to product features for a distributed (Multiplatform) installation

| Topology pattern | Number of clusters | Description |
|---|--------------------|---|
| Single Cluster | 1 | <p>Messaging, application deployment target, and application support functions are contained in a single cluster. This topology pattern is useful for synchronous messaging, proof of concept, or application testing environments.</p> <p>A Single Cluster topology pattern is ideal for limited hardware. Because all of the components are installed in the same cluster, fewer physical machines are required.</p> |
| Remote Messaging | 2 | <p>This topology pattern separates the messaging environment from the application deployment target and the application support functions. Use this topology pattern when message throughput is a critical requirement for your daily operation. This topology pattern is highly recommended for asynchronous messaging and transactional systems.</p> |
| Remote Messaging and Remote Support | 3 | <p>This topology pattern separates messaging, Common Event Infrastructure (CEI), application deployment target and application support functions into distinct clusters. Most businesses can use this topology pattern to support their deployment environments as it is designed for performance and isolation of transactional processing from messaging and other support functions.</p> <p>This topology pattern is the default topology pattern for WebSphere ESB production environments.</p> |
| Remote Messaging, Remote Support, and Web | 4 | <p>This topology pattern defines one cluster for application deployment, one remote cluster for the messaging infrastructure, one remote cluster for supporting applications, and one for web application deployment.</p> |

Determining whether to create a standardized or customized network deployment environment

After profile creation, there are two approaches to configuring the servers, server clusters and product components that make up a network deployment environment. You can create a standardized network deployment environment based on IBM-supplied topology patterns, or you can create a customized network deployment environment, setting up the servers, server clusters and product components in a manner customized to your business processing needs.

Reasons to create a standardized network deployment environment

If the IBM-supplied topology patterns (packaged as templates in WebSphere ESB) address all or most of your business processing needs, use Deployment Environment wizard to create a standardized network deployment environment. The Deployment Environment wizard generates clusters and servers according to a number of predefined topologies, and configures multiple components across them all at the same time.

Other reasons for creating a standardized network deployment environment rather than a customized network deployment environment include the following:

- You do not have a lot of experience using the features and functions of the administrative console required to create and configure servers, sever clusters and IBM Business Process Manager components.
- You want to configure multiple components by stepping through a single wizard in the administration application.
- You want to import the database design file to provide the values for database related resource definitions. Refer to “Creating database design files by using the database design tool” on page 97 for information on creating the database design file.
- You have a predefined Deployment Environment which you can import into the current environment and customize if necessary.

Reasons to create a customized network deployment environment

If the complexities of your business processing needs are not sufficiently met by any of the IBM-supplied topology patterns, use the features and functions of the administrative console create a customized network deployment environment.

Other reasons for creating a customized network deployment environment rather than a standardized network deployment environment include the following:

- You are well-versed in using the administrative console to create deployment environments
- You understand the concepts and component architecture required in a multi-clustered environment.
- You want to configure any clusters or servers upon which the components will be deployed *before configuring* the WebSphere ESB components themselves.

Planning your database configuration

To plan your database configuration, you need to know which databases must be in place and configured in order to use the software, which components of WebSphere ESB you will use and their associated databases, the tasks required to administer the databases, and the security privileges of the database system that you are using.

Databases and your WebSphere ESB topology

A database configuration is part of the overall WebSphere ESB topology.

If you create the stand-alone development profile (qesb) during installation, the required databases are configured automatically. You do not need to design the database requirements.

If you create database tables manually, use the database design tool to create the SQL scripts, because the tool ensures that the generated SQL scripts are unique.

You can incorporate configuration information for the database into the profile creation process by one of the following methods:

- Referencing a database design file
- Setting database configuration parameters with the Profile Management Tool or the **manageprofiles** command-line utility

Regardless of how you choose to implement your database configuration, you must generate the SQL scripts as part of the profile creation process.

Before they configure the databases, the solution architect and database administrator must collaborate on the database topology to understand the best way to store database tables. For example, will the tables be stored in the same database as the common database? Or will the tables be stored in a separate database as a stand-alone profile? Separate databases might be helpful because they simplify the database configuration. You might also use separate databases to tune and manage the component databases separately from the common database in a stand-alone server environment.

For more information about the database design tool, see *Creating database design files using the database design tool*.

Choosing how and when to configure the common database

You can create the required database tables either before or after configuring WebSphere ESB. The important thing to remember is that the databases (including their tables, schemas, and so on) must exist before the WebSphere ESB servers try to use them.

You can create the common database before, during, or after you create the WebSphere ESB profile.

- Before you configure WebSphere ESB:
 - Edit and run the default scripts that come with WebSphere ESB. You can use the default scripts to create only the common database and Business Process Choreographer tables.
 - Use the design file that was created using the database design tool (DDT). See *Creating database design files by using the database design tool*.
- After you configure WebSphere ESB:
 - Use the Profile Management Tool to configure WebSphere ESB to work with the tables in the database as you create the profile. You can create and configure the database tables during profile creation, or delay creation and configuration until after the profile has been created. Use the Profile Management Tool to generate database scripts that you can use to create and configure the database tables. These generated scripts are ready to use. No editing is required.
 - Use the design file that you created using the database design tool (DDT). See *Creating database design files by using the database design tool*.

Supported database types

Choosing a database depends on your operating system and on the features that you will use with WebSphere ESB.

See Table 14 on page 37 for a list of the databases that are supported with WebSphere ESB.

WebSphere ESB packages JDBC drivers for DB2, Oracle, and SQL Server. For information about the JDBC drivers (including version and level information), see the web page *Detailed hardware and software requirements for IBM WebSphere Enterprise Service Bus V7.5*.

Note: You are responsible for providing JDBC driver levels outside of what is packaged with WebSphere ESB.

The DB2 Express database is included in the WebSphere ESB software, and can be installed and configured automatically when you install WebSphere ESB.

Important: Linux If you are installing DB2 Express as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. See Kernel parameter requirements (Linux) for a list of the kernel requirements. You can locate the current values by parsing the output of the `ipcs -l` command.

Note: Currently, there is a known limitation in DB2 Express installer related to the inclusion of national language (NL) strings in properties passed to it from the WebSphere ESB installer. The following values, which are passed to DB2 Express when it is being installed cannot have NL strings in them: Linux Windows

- Linux Instance user name and password: bpminst and bpminst1
- Linux Fenced user name and password: bpmfenc and bpmfenc1
- Linux Administration server (DAS) user name and Password: bpmadmin and bpmadmin1
- Windows Administrative user name and Password: bpmadmin and bpmadmin1

Each database is represented by a parameter *dbType* which is a character string. The *dbType* parameter is used as a parameter in **manageprofiles** command-line utility. The values of *dbType* for the supported databases are shown in Table 14.

Table 14. Supported database types, their associated dbType values and restrictions

| Supported database | dbType value | Restrictions and notes |
|---|-----------------------|--|
| DB2 Express | DB2_UNIVERSAL | Used as the default database type for a stand-alone profile. |
| DB2® Universal | DB2_UNIVERSAL | |
| DB2 Data Server | DB2_DATASERVER | Available for download from: 9.7 GA level Fixpacks |
| DB2 for z/OS | DB2UDBOS390 | If you are using DB2 for z/OS as your database management system, you must configure the database and database objects by using the createDB.sh script. The installation wizard is not able to create a database of this type. . |
| DB2 UDB for iSeries® (Toolbox) DB2 for i5/OS® (Toolbox) | DB2UDBISERIES_TOOLBOX | This is the default database type used for network deployment topologies. |

Table 14. Supported database types, their associated dbType values and restrictions (continued)

| Supported database | dbType value | Restrictions and notes |
|----------------------|---|--|
| Microsoft SQL Server | Microsoft SQL Server JDBC 1.2 and 2.0 = MSSQLSERVER_MICROSOFT | Microsoft SQL Server JDBC 3.0 is also supported, though not listed as a separate database type on the Profile Management Tool. If you are using Microsoft SQL Server JDBC 3.0 as your database management system, selecting a database type of Microsoft SQL Server JDBC 2.0 will support version 3.0 Note: If a locale different from Latin must be specified, then the createDatabase.sql script can not be used. A different locale that is case-insensitive must be specified. Note: The databases that are created for the components must be case-sensitive. If you are using the SQL files to create the database for CommonDB, they create a case-sensitive database. Important: You must configure XA transactions after the database is installed and before you start the server. Failure to configure the XA transactions can result in an error during server start up. See "Configuring XA transactions" on page 92. |
| Oracle | ORACLE | The installation wizard is not able to create a database of this type for Oracle. |

Important: On i5/OS, there is a single global database in which you define all schemas for all functional components. You must make sure that all schema names are unique within the logical partition (LPAR).

A second parameter used in file path and file naming conventions is *feature*, which indicates which of the various databases is under consideration. Table 15 lists the databases and the associated *feature* parameter.

Table 15. Databases and their associated feature name.

| Database | Feature |
|--|--------------------|
| Business Space | BusinessSpace |
| Common database | CommonDB |
| Enterprise service bus logger mediation database | EsbLoggerMediation |

When you install WebSphere ESB, database scripts are created in the following locations:

INSTALL_ROOT/dbscripts/feature/dbType

Where *feature* can be:

- BusinessSpace
- CommonDB
- EsbLoggerMediation

Database naming restrictions

Databases cannot be reused across multiple installations of WebSphere ESB. Each installation of WebSphere ESB requires exclusive use of its associated databases. You must configure the databases so that they can be uniquely identified.

Depending on the installation path that you select, the databases associated with an installation might be configured with default names. For example, the default value for the Common Database associated with WebSphere ESB on IBM DB2 is CMNDB.

If you have two installations of WebSphere ESB that use DB2, you must select, for one of the installations, an installation path that lets you specify the names instead of accepting the default values.

To ensure the uniqueness of the database names, select installation paths that prompt you for the database names.

For example, you are prompted for the database names when you use the Typical installation path. Select the option to use an existing database server instead of the default DB2 Express.

Note: When you use the Profile Management Tool to create a profile after installation, you are prompted for database names, no matter which path in the Profile Management Tool (Typical or Advanced) you choose. The only exception is when you use a database design file for your database configuration. The database design file contains the database names, user name, and password information. Thus, the Profile Management Tool does not prompt you for this information.

In contrast, you are not prompted for database names in the following cases:

- When doing a GUI installation for the WebSphere ESB image, Installation Manager provides an option to create the profile, but assigns default values to the database names.
- You are using the Typical installation path and you choose to install DB2 Express, which is packaged with the installation images. Database names are assigned the default values.

Additional restrictions apply to database naming. These restrictions depend on the database server that you are using.

For Microsoft SQL Server database, databases must be case-sensitive.

Data sources for WebSphere ESB

Data sources provide a link between applications and relational databases. The data sources that you use are affected by whether you set up a stand-alone environment or a network deployment environment.

Applications use a data source to obtain connections to a relational database. A data source is analogous to the Java EE Connector Architecture (JCA) connection factory, which provides connectivity to other types of enterprise information systems (EIS).

A data source is associated with a Java Database Connectivity (JDBC) provider, which supplies the driver implementation classes that connect with a specific type of database. Application components interact directly with the data source to obtain connection instances to your database. The connection pool that corresponds to each data source provides connection management.

You can create multiple data sources with different settings, and associate them with the same JDBC provider. For example, you might use multiple data sources to access different databases within the same database application. In WebSphere ESB, JDBC providers must implement one or both of the following data source interfaces. Use these interfaces to run the application in a single-phase or two-phase transaction protocol.

ConnectionPoolDataSource

A data source that supports application participation in local and global transactions, except two-phase commit transactions. When a connection pool data source is involved in a global transaction, the transaction manager does not provide transaction recovery. The application is responsible for providing the backup recovery process if multiple resource managers are involved.

XADataSource

A data source that supports application participation in any single-phase or two-phase transaction environment. When this data source is involved in a global transaction, the WebSphere Application Server transaction manager provides transaction recovery.

The following tables provide examples of typical stand-alone environment setups and typical deployment environment setups:

Table 16. Typical stand-alone environment setup

| Datasource | Component | Scope | JNDI Name |
|------------------------------------|-----------|--------|--|
| WBI DataSource | CommonDB | Node | jdbc/WPSDB |
| SCA Application Bus ME data source | SCA ME | Server | jdbc/com.ibm.ws.sib/nlNode01.server1-SCA.APPLICATION.localhostNode01Cell.Bus |
| event | CEI | Server | jdbc/cei |
| CEI ME data source | CEI ME | Server | jdbc/com.ibm.ws.sib/nlNode01.server1-CEI.cellName.BUS |

Table 17. Typical deployment environment setup

| Datasource | Component | Scope | JNDI Name |
|------------------------------------|-----------|---------|--|
| WBI DataSource | CommonDB | Cell | jdbc/WPSDB |
| SCA Application Bus ME data source | SCA ME | Cluster | jdbc/com.ibm.ws.sib/clusterone-SCA.APPLICATION.enduranceTestCell01.Bus |
| event | CEI | Cluster | jdbc/cei |
| CEI ME data source | CEI ME | Cluster | jdbc/com.ibm.ws.sib/clusterone-CEI.cellName.BUS |

JDBC drivers and locations

The following tables list the supported JDBC drivers. The first table contains the names and locations of the JDBC drivers that are provided with the product. The second table contains the names of the JDBC drivers that are supported but not provided with the product.

The following supported JDBC drivers are included with the product installation files.

Table 18. Supported JDBC drivers and locations that are provided with the product

| Server | Driver description | Driver location | Comments |
|------------|--|---------------------------------|--|
| DB2 | IBM DB2 Universal JDBC Driver 3.61.65 | WAS_HOME/jdbcdrivers/DB2 | IBM DB2 Universal JDBC Driver is the default DB2 driver for both distributed and z/OS platforms. |
| | IBM Data Server Driver for JDBC and SQLJ 4.11.69 | | |
| Oracle | Oracle JDBC Driver 11g 11.2.0.1.0 | WAS_HOME/jdbcdrivers/Oracle | |
| SQL Server | Microsoft SQL Server JDBC Driver 2.0 | WAS_HOME/jdbcdrivers/SQL Server | Microsoft SQL Server JDBC Driver 2.0 supports SQL Server 1.2, 2.0, and 3.0. |

The following supported JDBC drivers are not included with the product installation files.


Table 19. Supported JDBC drivers that are not provided with the product

| Server | Driver description |
|------------|--------------------------------------|
| Oracle | Oracle JDBC Driver 11g 11.1.0.6 |
| SQL Server | Microsoft SQL Server JDBC Driver 1.2 |
| | Microsoft SQL Server JDBC Driver 3.0 |

Related reference:

manageprofiles parameters

Related information:

 Detailed hardware and software requirements for IBM Business Process Manager Advanced V7.5

Identifying required database administrator tasks

If you want to perform some types of database creation and configuration tasks in WebSphere ESB, you must be a database administrator (DBA).

Database selection

Choosing how to configure your database

Database privileges and security considerations

- “Database privileges” on page 43
- Identifying necessary security authorizations

Profile creation

- Prerequisites for creating or augmenting profiles
- Creating a stand-alone environment
- Configuring the software after a Custom installation to create one or more Deployment manager and Custom (managed node) profiles
-

Tip: If you use the deployment environment feature, you can use a database other than the default database server as your database product. The user ID

that you provide for the **User name to authenticate with the database** field on the database configuration panels must have DBA privileges.

Database configuration

- Create the database and tables before profile creation or augmentation
 - Creating database design files by using the database design tool
- Create the database and tables after profile creation or augmentation
 - Creating the Common database and tables after profile creation or augmentation
 - Creating database design files by using the database design tool
- “Planning to configure the messaging engine database” on page 55

Relevant links

- Configuring Business Space

Nonadministrative user considerations

If you are installing WebSphere ESB as a nonadministrative or nonroot user and you want to create a test profile during installation, you must have the DB2 server installed before you begin the installation. Remember the database details so that you can enter them during the installation.

The considerations described in this topic apply to any install scenario where you choose to install using the **Typical** install option. Profiles are created automatically when you install using the **Typical** option.

To install as a nonadministrative user, you have the following choices:

- Before installing the product, install a DB2 server separately. For information about installing DB2 as a nonadministrative or nonroot user, see [Linux](#)
[UNIX](#) [Windows](#)
 - [Linux](#) [UNIX](#) Non-root installation overview (Linux and UNIX)
 - [Windows](#) Required user accounts for installation of DB2 server products (Windows)
- Logon as an administrator and use the product installer to install the DB2 server alone. Grant special permission to the nonadministrative user. Then logon as the nonadministrative user and install the product using the installed DB2 server.

Alternatively, instead of creating a test profile, you can create a profile after installation . Use these steps:

1. Install the product without creating a profile. When you install as a nonadministrative user, on the Install Packages page, you must clear the check box for DB2 Express. On Windows, if you have the option to install IBM Cognos Business Intelligence, you must clear that check box as well.
2. On the Features page, expand the servers and make sure that none of the test profiles are selected.
3. Use the Profile Management Tool to create a stand-alone profile, or to create the deployment manager and the custom profiles. If you do not have a database installed, use the **Advanced** path for all. Do not use the **Typical** path. Select the option to delay the execution of the database scripts during profile creation.
4. If the databases were not created in advance. have the database administrator create the databases and tables after profile creation or augmentation.
5. For a network deployment:

- a. Federate the custom profiles to the deployment manager.
- b. Using the administrative console, create the required deployment environment

Note: If you choose to use the DB2 Express database included (and optionally installed) with the product, you must meet the following criteria:

- Uninstall any other versions of DB2 from the system
- Install IBM Business Process Manager as a nonadministrative or nonroot user

Database privileges

Set database privileges to determine the authority that you must have to create or access your data store tables for each supported database management system.

When you create schemas with the installer, Profile Management Tool, database design tool, or scripts, your user ID must have the authority to create tables. When the tables are created, you must have the authority to select, insert, update, and delete information in the tables.

The following table describes the database privileges that are needed to access the data stores.

Table 20. Database privileges

| Header | Minimum privileges required to create objects in the database | Minimum privileges required to access objects in the database |
|--------------|--|---|
| DB2 | The user ID needs CREATETAB authority on the database and CREATETS to create the table space. The user ID also needs CREATEIN and DROPIN privilege on the schema. The user ID needs system privileges CREATEDBA and CREATEDBC. The user ID also needs ALTER, DELETE, INDEX, INSERT, REFERENCES, SELECT, and UPDATE privileges on the created tables. | The user ID needs SELECT, INSERT, UPDATE, and DELETE privileges on the tables. The user ID also needs EXECUTE ON PROCEDURE on stored procedures. Refer to Table 21 on page 44 for detailed DB2 database privileges for WebSphere ESB and WebSphere Enterprise Service Bus components. |
| DB2 for z/OS | The user ID needs CREATETAB authority on the database and CREATETS to create the table space. The user ID also needs CREATEIN and DROPIN privilege on the schema. To create storage groups for the database, the user ID needs CREATESG, CREATEDBA, and CREATEDBC system privileges. The user ID also needs ALTER, DELETE, INDEX, INSERT, REFERENCES, SELECT, and UPDATE privileges on the created tables. | The user ID needs SELECT, INSERT, UPDATE, and DELETE privileges on the tables. The user ID also needs EXECUTE ON PROCEDURE on stored procedures. Refer to Table 22 on page 45 for detailed DB2 for z/OS database privileges for WebSphere ESB and WebSphere Enterprise Service Bus components. |

Table 20. Database privileges (continued)

| Header | Minimum privileges required to create objects in the database | Minimum privileges required to access objects in the database |
|------------|--|--|
| Oracle | The user ID needs sufficient privilege to create relational tables and indexes in the data store schema. The database also needs a space quota in the default table space of the owner of that schema. | <p>The user ID needs the SESSION privilege to connect to the database. If the same user ID owns both the data store schema, and the component that is connecting to the database, the user ID has sufficient privilege to manipulate the tables. Otherwise, the user ID needs SELECT, INSERT, UPDATE, ALTER, and DELETE object privileges on the tables that make up the data store, and the DROP ANY TABLE system privilege to enable the use of the TRUNCATE TABLE statement. The user ID also requires CREATE INDEX and INDEXTYPE privileges.</p> <p>You must create the Oracle database using a UTF-8 character set, which supports the other customer character sets that are supported by WebSphere ESB.</p> <p>See Table 23 on page 45 for detailed Oracle database privileges for WebSphere ESB and WebSphere Enterprise Service Bus components.</p> |
| SQL Server | The user ID ideally requires DB OWNER privileges on the data stores used for WebSphere ESB. | <p>Configure the SQL Server for SQL Server and Windows authentication so that authentication to be based on an SQL server login ID and password. The user ID must be the owner of the tables, or a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.</p> <p>See Table 24 on page 45 for detailed SQL Server database privileges for WebSphere ESB and WebSphere Enterprise Service Bus components.</p> |

Table 21 describes additional DB2 database privileges for WebSphere ESB components.

Table 21. Detailed DB2 database privileges

| Component | Installation privileges | Runtime privileges |
|-------------------|--|---|
| Common DB | CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE | SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE |
| Business Space | CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE | SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE |
| Messaging Engines | CREATE TABLE, CREATE INDEXTYPE | SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE |

Table 22 describes additional DB2 for z/OS database privileges for WebSphere ESB components.

Table 22. Detailed DB2 for z/OS database privileges

| Component | Installation privileges | Runtime privileges |
|-------------------|--|--|
| Common DB | CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE | SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, USAGE ON SEQUENCE |
| Business Space | CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE | SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, USAGE ON SEQUENCE |
| Messaging Engines | CREATE TABLE, CREATE INDEXTYPE | SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE |

Table 23 describes additional Oracle database privileges for WebSphere ESB components.

Important: If you configure all the following components for a single Oracle database, you can create a superset of all the privileges that are specified for each component. If you configure the four components for numerous databases, you can set different privileges for each.

Table 23. Detailed Oracle database privileges

| Component | Installation privileges | Runtime privileges |
|-------------------|--|---|
| Common DB | CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE | SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE |
| Business Space | CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE | SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE |
| Messaging Engines | CREATE TABLE, CREATE INDEXTYPE | SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE |

Table 24 describes additional SQL Server database privileges for WebSphere ESB components.

Table 24. Detailed SQL Server database privileges

| Component | Installation privileges | Runtime privileges |
|-----------|--|---|
| Common DB | CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE | SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE |

Table 24. Detailed SQL Server database privileges (continued)

| Component | Installation privileges | Runtime privileges |
|-------------------|--|---|
| Business Space | CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE | SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE |
| Messaging Engines | CREATE TABLE, CREATE INDEXTYPE | SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE |

Business Process Choreographer Explorer reporting function is not supported on SQL Server.

For more information, see the WebSphere Application Server page in the related reference.

User ID or schema name privileges:

During the installation of WebSphere ESB, you can use the default schema name and user ID privileges to install your databases. However, your database design might require separate user ID or schema name privileges.

Review the provided scenarios to determine when and how to configure different schema names and user ID privileges when you install WebSphere ESB.

Scenario for a single user ID or schema name privileges

If you chose a default installation for your databases, WebSphere ESB requires a minimum of one user ID or schema name that can create tables and to select, insert, update, and delete rows in those tables. You can use the Profile Management Tool or the installer to create your databases.

The following table shows the default database configuration properties when you use DB2 as your database. Other databases have different default configuration properties for database configuration.

Table 25. Scenario: Single user ID or schema

| Database tables | Default database name with DB2 | User ID or schema name |
|------------------------|--------------------------------|---|
| Common database tables | CMNDB | WebSphere ESB provides a user ID during installation. |
| Messaging tables | MEDB | WebSphere ESB provides a schema name during installation. |

If your database design has different properties, you might need multiple user ID and schema name privileges. The following scenarios show you how to apply the configuration to achieve your desired design. Even if your particular design is not in the provided scenarios, you can adapt some of the ideas to implement your particular design.

Scenario 1 for multiple user ID or schema name privileges

In this scenario, you use a schema name that is the same as the user ID privileges, but you do not use the default schema name or default user ID privileges. This

single user ID can access all of the database and also create all needed tables. The following examples show scenario 1 privileges:

- Schema name: dog
- Schema name for SCA.SYSTEM ME : dogSYS
- Schema name for SCA.APP ME: dogAPP
- Schema name for Event ME: dogEvent
- User ID to create schemas: dog
- User ID to select, insert, update, and delete schemas: dog

The following table contains information about how to set up the schema name and user ID privileges with DB2 as your database. If you choose a different database, see their documentation for setting up schema names and user ID privileges.

Table 26. Scenario 1: Multiple user ID or schema

| Database tables | Database name with DB2 | Schema name | User ID to create tables | User ID to select, insert, update, and delete rows |
|------------------------|--|--|--|--|
| Common database tables | You supply this value in the <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation | This schema name is the same as the user ID that is used to select, insert, update, and delete rows. | This value is the same as the user ID that is used to select, insert, update, and delete rows. | You supply this value in the <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation |

Scenario 2 for multiple user ID or schema name privileges

In this scenario, you use the same schema name and user ID to select, insert, update, and delete schemas. However, you use a different user ID to create the schemas. The following examples show scenario 2 privileges:

- Schema name: snow
- Schema name for SCA.SYSTEM ME: snowSYS
- Schema name for SCA.APP ME: snowAPP
- Schema name for Event ME: snowEvent
- User ID to create the schemas: rock
- User ID to select, insert, update, and delete schemas: snow

The following table contains information about how to set up the schema name and user ID privileges with DB2 as your database. If you choose a different database, see their documentation for setting up schema names and user ID privileges.

Table 27. Scenario 2: Multiple user ID or schema

| Database tables | Database name with DB2 | Schema name | User ID to create tables | User ID to select, insert, update, and delete rows |
|------------------------|---|---|---|---|
| Common database tables | <p>You supply this value twice:</p> <ol style="list-style-type: none"> 1. In table creation scripts 2. During the WebSphere ESB configuration with one of the following: <ul style="list-style-type: none"> • Administrative console • Installation wizard • Profile Management Tool • Silent install • Silent profile creation <p>Restriction: If you execute the installer first, then you supply the value once because the generated scripts already contain the correct schema name and user ID values.</p> | The table creation scripts need to be modified with the schema name that allows reading and writing rows. | The table creation script needs to be modified with the user ID that allows table creation. | <p>You supply the user ID during profile creation through one of the following:</p> <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation |

Scenario 3 for multiple user ID or schema name privileges

In this scenario, you use the same user ID to create all schemas. However, each schema has a different user ID to select, insert, update, and delete rows. The following list shows examples of privileges for Scenario 3:

- Schema name: waterCom
- Schema name for common tables: waterCom
- Schema name for SCA.SYSTEM ME: waterSYSME
- Schema name for SCA.APP ME: waterAPPME
- Schema name for Event ME: waterEventME
- Schema name for ESBMessaging tables: waterESB
- User ID to create schemas: milk
- User ID to select, insert, update, and delete schemas:

| Schema name | User ID to select, insert, update, and delete schemas |
|--------------|---|
| waterCom | waterCom |
| waterSYSME | waterSYSME |
| waterAPPME | waterAPPME |
| waterEventME | waterEventME |
| waterESB | waterESB |

The following table contains information about how to set up the schema name and user ID privileges with DB2 as your database. If you choose a different database, see their documentation for setting up schema names and user ID privileges.

Table 28. Scenario 3: Multiple user ID or schema

| Database tables | Database name with DB2 | Schema name | User ID to create tables | User ID to select, insert, update, and delete rows |
|------------------------|--|--|--|--|
| Common database tables | You supply this value in the <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation | This schema name is the same as the user ID that is used to select, insert, update, and delete rows. | This value is the same as the user ID that is used to select, insert, update, and delete rows. | You supply the user ID during profile creation through one of the following: <ul style="list-style-type: none"> • Installation wizard • Profile Management Tool • Silent install • Silent profile creation |
| Messaging tables | You supply this value with the definition of each messaging engine. | The table creation scripts must include the schema name that is used to select, insert, update, and delete rows. | This value is the same as the user ID that is used to select, insert, update, and delete rows. | You supply this value during the creation of the messaging engine. Select the Create Table option during the messaging engine configuration. |

Planning your component-specific database configurations

WebSphere ESB includes components that require database tables and specific names of the databases where the tables are stored.

Use the information in this section to familiarize yourself with WebSphere ESB components that your database administrator must manage, configure, and administer.

To plan your database configuration, you must know the components that you will use. Table 29 on page 50 lists the WebSphere ESB components that require a database table, and the default names of the databases where the tables associated with these components are stored.

Important: You can change these names if you choose, but you must remember to use the names consistently in later configuration steps.

Table 29. Databases that are required by individual components

| Server component | Database (default name) | Notes |
|-----------------------------------|-----------------------------|--|
| Business Space | CMNDB (the common database) | For stand-alone profiles, you must create the common database before you start WebSphere ESB. For other profiles, you must use the administrative console to configure Business Space. Configuring a Business Space database is mandatory for using Business Space powered by WebSphere, which provides a common interface for application users to create, manage, and integrate web interfaces across a range of IBM products. |
| Common Event Infrastructure (CEI) | EVENT (stores events) | <p>CEI database configuration is not supported by the Profile Management Tool or the manageprofile command-line utility.</p> <p>Do not create this database for production environments because the performance of persisting events may be impacted.</p> <p>The Common Base Event browser relies on the CEI database. If you want to use the Common Base Event browser to retrieve and view logging, tracing, management, and business events in your business enterprise applications, you must create the CEI database manually.</p> |
| Enterprise service bus | CMNDB (the common database) | You must configure these tables either during startup of the deployment manager or stand-alone server or before you start the deployment manager or stand-alone server |
| Relationships | CMNDB (the common database) | You must create the common database before you start WebSphere ESB. You must configure the CMNDB tables before or during the startup of the deployment manager or stand-alone server. |
| SIBus | User created | You must configure these tables during the startup of the messaging engine or before you start the messaging engine. You can use a file store with SIBus in a stand-alone environment during profile creation. However, you cannot use a file store with SIBus in a network deployment environment. |

Planning to configure the common database:

The common database configurations contain information about supported database types, script names and their locations, profile creation configuration actions, installation parameters, types of created tables, and user ID privileges.

The WebSphere ESB common database is used by the following product components:

- Relationship service
- Enterprise Service Bus (ESB) Logger Mediation Primitive

You can create the common database before, during, or after you create the WebSphere ESB profile.

- Before you configure WebSphere ESB:
 - Edit and run the default scripts that come with WebSphere ESB. You can use the default scripts to create only the common database and Business Process Choreographer tables.
 - Use the design file that was created using the database design tool (DDT). See Creating database design files by using the database design tool.
- After you configure WebSphere ESB:
 - Use the Profile Management Tool to configure WebSphere ESB to work with the tables in the database as you create the profile. You can create and configure the database tables during profile creation, or delay creation and configuration until after the profile has been created. Use the Profile Management Tool to generate database scripts that you can use to create and configure the database tables. These generated scripts are ready to use. No editing is required.
 - Use the design file that you created using the database design tool (DDT). See Creating database design files by using the database design tool.

Supported database types

The common database can use the following database products:

Table 30. Supported database products

| Database Types | Considerations |
|------------------------------------|--|
| DB2 Express® | Used as the default database type for a stand-alone profile. |
| DB2 Universal | Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |
| DB2 Data Server | Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |
| DB2 for z/OS v8 DB2 for z/OS v9 | Important: When creating a profile for a server that uses DB2 for z/OS v9, the server must be able to connect to the DB2 database. Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |
| Microsoft SQL Server (Microsoft) | |
| Oracle | You need system database administrator privileges to create the database, tables, and schemas. If you do not have these privileges, you might receive errors when you create or access the tables and schemas. |

User ID privileges

The user credentials that you provide in the Profile Management Tool must have the permissions necessary to create table spaces, tables, schemas, indexes, and stored procedures. For the **Create new database** option, the user ID must have the necessary privileges to create a database. If the user who is running the script has the authority to create tables, the script does not require an authentication ID within the script. For more information, see “Users and schemas for databases” and “Database privileges”.

Database Management Service instances

For a network deployment environment, there is one set of common database tables per cell.

Configuration actions during profile creation

You can use one of the following options to install the common database:

- Installer
- Profile Management Tool
- Silent installation
- Scripts

Within each of these options are several more choices.

Installer

Use the Installer if you are going to create your profiles when you install your software. You can install your database products during installation, but you are limited in the types of database products that you can install. To use a supported database product that is not in Table 31, you must use the Profile Management Tool to create your deployment manager.

Table 31. Installer options

| Option | Databases you can use |
|--|--|
| Stand-alone development profile (qesb), created during software installation | DB2 Express |
| Customized: stand-alone profile | <ul style="list-style-type: none">• DB2 Universal• DB2 Data Server• Oracle |
| Customized: all other profiles | <ul style="list-style-type: none">• DB2 Universal• DB2 Data Server• Oracle |

If you create a stand-alone development profile (qesb), default values are used for configuration parameters and you cannot change these defaults. If you choose a customized installation, you can change the defaults for your specific requirements.

Profile Management Tool

Use the Profile Management Tool to create profiles after you install your software. The Profile Management Tool allows you the options of installing your database before, during or after profile creation.

You can use any of the “Supported database types” on page 51

Silent Installations

When you install the product silently, you can specify the common database configuration by editing the template response file.

Scripts

You can use scripts to create your common database before you install WebSphere ESB or during profile creation.

If you choose to configure your database manually after profile creation, you must first install WebSphere ESB and indicate in the Profile Management Tool that you do not want to run the scripts as part of profile creation. The Profile Management Tool updates the default scripts with the database parameters that you specify, and writes updated scripts out to the *profile_root/dbscripts/CommonDB/dbType/dbName* directory.

Tip: You can use the Profile Management Tool to change the directory to which updated scripts are written.

The scripts are ready to run, but you can edit them to include any specific requirements. You can then give these scripts to the person who should create your common database. If you try to start WebSphere ESB before creating the database, you receive an error message.

When you run the scripts, you also perform the following tasks:

- Create a database, if appropriate (valid only for a local database), depending on your choices in the Database configuration panel in the Profile Management Tool.

Important: Although you can defer creating the database until the profile creation is complete, you must enter valid information in the Database configuration panel of the Profile Management Tool. This information is used to create the data source for the WebSphere ESB.

- Create the data source on the JDBC provider.

Important: If you create the database at the same time as the profile, and if you introduce mistakes in the database parameters, errors occur in the profile. If you delay creating the database, the profile is created without errors, but the generated database scripts contain errors and you must correct them before you can create the database. For custom (managed) nodes of a cell, you must select the same database type as the deployment manager profile. The data source is maintained only at the cell level.

Note: For custom (managed) nodes of a cell, you must select the same database type as the deployment manager profile. The data source is maintained only at the cell level.

SQL scripts

Use SQL scripts to configure your database before or after you create the profile. Tables are created with a deployment manager profile so no SQL scripts are executed while the managed node is created.

You can find SQL scripts for each common database client in the following locations:

- *media_root/dbscripts* on your product media
- *install_root/dbscripts/CommonDB/dbType* after you install WebSphere ESB

If you choose to defer creation of the database after you create the profile, you can find the updated scripts in the *profile_root/dbscripts/feature/dbType/dbName* directory.

The SQL script naming convention is:

- For a component-specific script: `createTable_componentName.sql`, for example `createTable_Recovery.sql`
- For a component-independent script: `createTable.sql`.

The following table shows the script naming convention.

Table 32. Common database script naming convention

| Type of script | Script name |
|-----------------------|---|
| Component specific | <code>scriptName_componentName.sql</code> |
| Component independent | <code>scriptName.sql</code> |

JDBC provider

A new Java Database Connectivity (JDBC) provider is created depending on the database type. The provider is created in the node scope in a stand-alone profile and at the cell level in a network deployment environment. The JDBC provider refers to the `JDBC_DRIVER_PATH` variable to locate local JDBC drivers. The variable is specified at the cell level and each node level points to the correct local path.

Data source name:

- WPS DataSource

Data source JNDI name:

- jdbc/WPSDB

Restrictions

Several restrictions exist for the database commands that are available during profile creation.

Create new database is disabled for the following database types:

- DB2 for z/OS
- Oracle
- Microsoft SQL Server

Tables

The common database scripts create only static tables during profile creation. The following table contains a list of all the tables that are created by different components.

Table 33. Tables created by WebSphere ESB components

| Component | Table names | Scripts |
|-----------------|-----------------------------------|--|
| Relationship | Dynamic table, created at runtime | <code>createTable_RelationshipMetadataTable.sql</code> |
| Common database | SchemaVersionInfo | <code>createTable_CommonDB.sql</code> |

Table 33. Tables created by WebSphere ESB components (continued)

| Component | Table names | Scripts |
|----------------------|-------------|--|
| ESB Logger Mediation | MSGLOG | createTable_ESBLogger Mediation.sql |

All the SQL scripts in the previous table are executed by the `commonDBUtility.ant` file from each component script, such as **configRecovery > commonDBUtility > execute createTable_Recovery.sql**. When the value `delayConfig=true` is in the response file, the SQL files are created, but they are not run. In this case, you must run the SQL manually after the configuration.

In the WebSphere Enterprise Bus Logger Mediation component, you can configure each message logger primitive to use a different data source and a different database.

Exported scripts

Scripts are created for any option that you selected on the Profile Management Tool panel to configure the common database. The scripts contain only basic creation statements for databases, tables, and indexes. The database administrator must use database native commands to execute these scripts. For more information, see "Configuring the common database using the Profile Management Tool".

The names of the scripts are `configCommonDB.bat` for Windows, and `configCommonDB.sh` for UNIX-based operating systems.

Database scripts are exported to the

`profile_root/dbscripts/CommonDB/dbType/dbName` directory.

Planning to configure the Common Event Infrastructure database:

The Common Event Infrastructure (CEI) database specifications list the types of supported databases, script locations, profile configuration types, and necessary user ID privileges. You can optionally use the CEI database to store events that are captured when it is monitoring WebSphere ESB.

The CEI database is an internal device and you do not interact directly with it. You must use the supported CEI programming interfaces for all interactions with the CEI database.

When you run the Profile Management Tool, you do not automatically create the CEI database. If you want to store CEI events, you must create the database manually for a stand-alone profile and for each instance of a CEI server in a network deployment environment.

Planning to configure the messaging engine database:

The messaging engine database specifications list supported database type, scripts and their locations, profile creation types, and necessary user ID privileges.

The messaging engine database is used to store operating information. Essential objects that the messaging engine needs for recovery in the event of a failure are also stored.

The messaging engine database is used by the message engines for Service Component Architecture (SCA) modules and Common Event Infrastructure (CEI). The default database name for the SCA messaging engine is SCADB. For the other messaging engines, the default database name is MEDB. The default schema name is IBMWSSIB.

Important: Multiple schemas are not supported by all database types. For more information, see your database documentation.

In a stand-alone environment, you can use the administrative console to configure your SCA messaging engine. In a patterned network deployment environment, the messaging engines are configured during deployment environment creation. For a custom network deployment environment, you need to configure the messaging engines manually.

You have control over the messaging engine databases. For example, you can create a database for each messaging engine or you can use a single database for all the messaging engines. Each messaging engine must have either its own database or a schema.

Supported database types

The messaging engine database can use the following database products:

Table 34. Supported database products

| Database Types | Considerations |
|------------------------------------|--|
| DB2 Express | Used as the default database type for a stand-alone profile. |
| DB2 Universal | Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |
| DB2 Data Server | Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |
| DB2 for z/OS v8 DB2 for z/OS v9 | Important: When creating a profile for a server that uses DB2 for z/OS v9, the server must be able to connect to the DB2 database. Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |
| Microsoft SQL Server (Microsoft) | |
| Oracle | You need system database administrator privileges to create the database, tables, and schemas. If you do not have these privileges, you might receive errors when you create or access the tables and schemas. |

User ID privileges

The user credentials that you provide in the Profile Management Tool must have the permissions necessary to create table spaces, tables, schemas, indexes, and stored procedures. For the **Create new database** option, the user ID must have the necessary privileges to create a database. If the user who is running the script has the authority to create tables, the script does not require an authentication ID within the script. For more information, see “Users and schemas for databases” and “Database privileges”.

For a network deployment environment, you need all necessary permissions for user privileges specified during configuration from the administrative console.

Important: For DB2 V9.7, grant the appropriate authority to the newly created user, because the user creation process does not automatically grant the user the necessary authority.

Database Management Service (DBMS) instances

Each messaging engine has its own database or schema:

- One is used to host each messaging engine for the Service Component Architecture system bus.
- Another is used to host each messaging engine for the Service Component Architecture application bus.
- Another is used to host each messaging engine for the Common Event Infrastructure bus.

The following list contains the naming conventions for the JDBC data source that the messaging engine uses to interact with the database:

- System bus: `<node><server>|<cluster>-SCA.SYSTEM.<cell>.Bus`
- Application bus: `<node><server>|<cluster>-SCA.APPLICATION.<cell>.Bus`
- CEI bus: `<node><server>|<cluster>-CEI.cellName.BUS`

Configuration actions during profile creation

Network deployment

No messaging engine databases are created automatically. After the profile is created, you can use the Configure your Network Deployment Environment guided activity to configure a server or a cluster for SCA. To access this guided activity from the administrative console of the deployment manager, expand **Guided Activities** and click **Configure your Network Deployment Environment**.

You can view the SCA configuration of your server from the **Application servers > servername > Service Component Architecture** panel of the administrative console.

The following administrative tasks are performed during profile creation:

- Remote Destination Location:
 - `configSCAAsyncForServer`, `configSCAJMSForServer` (`remoteMELocation` is true)
 - `configSCAAsyncForCluster`, `configSCAJMSForCluster` (`remoteMELocation` is true)
- Local Destination Location:
 - `configSCAAsyncForServer`, `configSCAJMSForServer`
 - `configSCAAsyncForCluster`, `configSCAJMSForCluster`

For more information about these tasks, see “`configSCAAsyncForCluster` command” and “`configSCAAsyncForServer` command”.

When you perform asynchronous SCA configuration for a server or cluster, a messaging engine is created for the SCA system bus. When you perform the JMS element of the SCA configuration for a server or cluster, a messaging engine is created for the SCA application bus. For both messaging engines, you must create a database or schema.

To configure the Common Event Infrastructure messaging engine, use the `deployEventService` administrative task to configure the event server and the Common Event Infrastructure bus.

SQL scripts

No SQL scripts are created as part of the product. You can use existing base WebSphere Application Server scripts to create database and tables if necessary. To create the MEDB manually before it is configured, use the **Application servers** > *servername* > **Service Component Architecture** panel of the administrative console.

JDBC provider

Service Component Architecture

The JDBC provider is reused when the JDBC provider implementation class has to match with the one chosen in the advanced configuration. If the same database types are used, then the implementation classes usually match. If no matching JDBC provider is found in the `resource.xml` file, then the `jdbc-resource-provider-templates.xml` file in the `templates/system` directory (profiles configuration) is searched for a matching JDBC provider. The provider is matched also against the implementation class.

Common Event Infrastructure

The JDBC provider creation for messaging engine database is similar to the approach followed in the creation of the CEIDB database.

Data source names

- System bus: : `_(node.server|cluster)-SCA.SYSTEM.cell.Bus/cell/cluster/server/node`
- Application bus: `_(node.server|cluster)-SCA.APPLICATION.cell.Bus/cell/cluster/server/node`
- Common Event Infrastructure: `_(node.server| cluster-CEI.cellName.BUS/cluster/server/node`

Data source JNDI names

- System bus: `jdbc/com.ibm.ws.sib/(node.server|cluster)-SCA.SYSTEM.cell.Bus/cell/cluster/server/node`
- Application bus: `jdbc/com.ibm.ws.sib/(node.server|cluster)-SCA.APPLICATION.cell.Bus/cell/cluster/server/node`
- Common Event Infrastructure: `Jdbc/ com.ibm.ws.sib/(node.server|cluster)-CEI.cellName.BUS/cluster/server/node`

Restrictions

There are no known restrictions.

Tables

For information on the tables, see the topic “Data stores” in the WebSphere Application Server Network Deployment information center.


Exported scripts

You can use the **sibDDLGenerator** script in `WAS_INSTALL_ROOT/bin` to create the SQL scripts for messaging engines database. Use the **sibDDLGenerator** script for creating

SQL scripts for use in production environment especially on the DB2 for z/OS platform. For more information, see the “The sibDDLGenerator command”.

These scripts contain only basic create database/tablespace/table statements. A database administrator might still need to tailor these scripts to meet their database needs, especially on DB2 for z/OS.

Related information:

 Configuring messaging engine and server behavior when a data store connection is lost

Planning to configure the logger mediation database tables for WebSphere Enterprise Service Bus:

You can find logger mediation database table specifications for WebSphere ESB. The specifications contain information about supported database types, script names and their locations, profile creation configuration actions, schema upgrades, and user ID privileges.

The logger mediation database tables are used by the Message Logger mediation primitive in WebSphere ESB. The Message Logger primitive stores message information in the common database. The common database is the default for the WebSphere ESB logger mediation database, but you can use an external database. During the profile augmentation phase, the system creates the `ESB_MESSAGE_LOGGER_QUALIFIER` variable, which is set to the value of the chosen common database schema qualifier.

The database is created automatically for a stand-alone configuration. You can use Data Definition Language (DDL) files to use additional databases for a stand-alone server configuration or for a network deployment environment.

For a stand-alone configuration that uses a DB2 for z/OS database, or for a managed node or deployment manager in a network deployment configuration, you must create the WebSphere ESB database and storage groups first. Then you can run the configuration script for WebSphere ESB for z/OS.

Supported database types

The WebSphere ESB logger mediation database can use the following database products:

Table 35. Supported database products

| Database Types | Considerations |
|-----------------|--|
| DB2 Express | Used as the default database type for a stand-alone profile. |
| DB2 Universal | Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |
| DB2 Data Server | Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |

Table 35. Supported database products (continued)

| Database Types | Considerations |
|------------------------------------|--|
| DB2 for z/OS v8 DB2 for z/OS v9 | Important: When creating a profile for a server that uses DB2 for z/OS v9, the server must be able to connect to the DB2 database. Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations. |
| Microsoft SQL Server (Microsoft) | |
| Oracle | You need system database administrator privileges to create the database, tables, and schemas. If you do not have these privileges, you might receive errors when you create or access the tables and schemas. |

User ID privileges

The user credentials that you provide in the Profile Management Tool must have the permissions necessary to create table spaces, tables, schemas, indexes, and stored procedures. For the **Create new database** option, the user ID must have the necessary privileges to create a database. If the user who is running the script has the authority to create tables, the script does not require an authentication ID within the script. For more information, see “Users and schemas for databases” and “Database privileges”.

Database Management Service (DBMS) instances

The common database is used for both stand-alone environments and network deployment environments at cell scope. However, you can manually create as many other instances as you require. Each message logger mediation primitive can be configured to use a different data source and therefore a different database.

Configuration actions during profile creation

For the stand-alone profiles and deployment manager profiles, the WebSphere ESB logger profile executes the **createTable** common database script in the common database.

Stand-alone profile

In a default stand-alone environment, a DB2 database named `EsbLogMedDB` is automatically created.

Network deployment environment

The default WebSphere ESB database is not automatically selected during network deployment profile creation. You must select the default database or one of the supported databases.

SQL scripts

The **createTable_ESB.sql** SQL script is located in the `install_root/dbscripts/CommonDB/DBTYPE` directory.

The **createMessageLoggerResource.jacl** and **removeMessageLoggerResource.jacl** scripts are located in the `install_root/bin` directory, and can be used to create or delete tables in the requested database type.

JDBC provider

The common database JDBC provider and data source are used by default:

Data source name:

- WPS DataSource

Data source JNDI name:

- jdbc/WPSDB

You can configure the Message Logger mediation to use a different data source.

Restrictions

There are no known restrictions.

Tables

The WebSphere ESB logger mediation database uses the MSGLOG table in the common database. However, you can choose not to use the common database, and use an external database instead.

Exported scripts

The database scripts are exported to the *install_root/dbscripts/CommonDB/DBTYPE/dbName* directory.

Schema upgrade scripts

No schema upgrade involved for MSGLOG table. When you migrate to WebSphere ESB V6.1, WebSphere ESB continues to use the MessageLogger databases used in prior releases. There is no support to migrate this data into the WebSphere ESB common database.

If you want to maintain a single location for message information, you can perform one of the following tasks:

- Manually move the data from the old database to the new database
- Continue to use the old database
- Use the **createMessageLoggerResource.jacl** script to move the data.

Planning error prevention and recovery

You can develop error-prevention and recovery strategies to minimize the impact of system and application errors.

Topics in *Planning error prevention and recovery* include links to a variety of resources, such as information center topics, technical articles and IBM Redbooks that provide detailed information on development processes and system configuration patterns designed to take advantage of WebSphere system recovery capabilities.

Overview of error prevention and recovery

The error prevention and recovery information describes how to avoid problems that might cause system failures, and provides or points to information on how to recover from system failures that can result from both ordinary and extraordinary circumstances.

WebSphere ESB is a middleware server optimized for enabling the running and management of business process management (BPM) and service-oriented architecture (SOA) solutions. WebSphere ESB is built on the foundational capabilities of WebSphere Application Server.

Middleware systems run under various conditions, not all of which are traditionally “good path” conditions. Many of the key features within WebSphere ESB are intended to deal with the uncertainty that might arise through what can appear to be normal operations.

Assumptions and expectations

Before using the information about system failure and recovery as described in the *Planning error prevention and recovery* section, read the following list of assumptions:

- You are familiar with WebSphere ESB and the basic architectural principles upon which it is built and the basic kinds of applications that it runs.
- You have a foundational understanding of integration projects, including how to plan for and implement integration projects.
- Unless otherwise specified, the information about system failure and recovery is relevant to version 6.1.0 and later of WebSphere ESB.

Note: The information contained in the *Planning error prevention and recovery* section assume a remote messaging and remote support pattern, which consists of three separate clusters, one for the WebSphere ESB and one each for the messaging engine and CEI event server.

Planning error prevention

As with all IT endeavors, planning against and practicing for extreme situations will increase the possibility for a successful recovery.

There are a number of required considerations associated with preparing for system and application recovery. These considerations can be grouped under two categories as follows:

- Error prevention practices as part of application design
- Error prevention practices as part of development process

Error prevention as part of application design:

Including error prevention practices as part of your application design means implementing specific design techniques and using the capabilities of the product to help prevent system and application errors.

A strong system of governance, complete with architectural and design guidelines and appropriate standards combined with reviews and checkpoints are essential to building the right kind of application.

Error prevention practices as part of application design include the following:

- Implementing design considerations for exceptions and faults

- Implementing an error handling strategy that uses existing WebSphere ESB error handling capabilities and tools
- Creating connectivity groups and using module application design techniques

Connectivity groups:

A connectivity group represents a specific pattern of behavior found in an SCA module.

Create connectivity groups to represent the possible request sources for the system.

In a connectivity group you:

- Put all the logic to get the inbound data into one module
This is also true for outbound data when it is going to an external system or legacy system
- Put all the logic to connect and transform the data into one module
All the other modules can now use a standard set of interfaces and not have to worry about extra transformations.

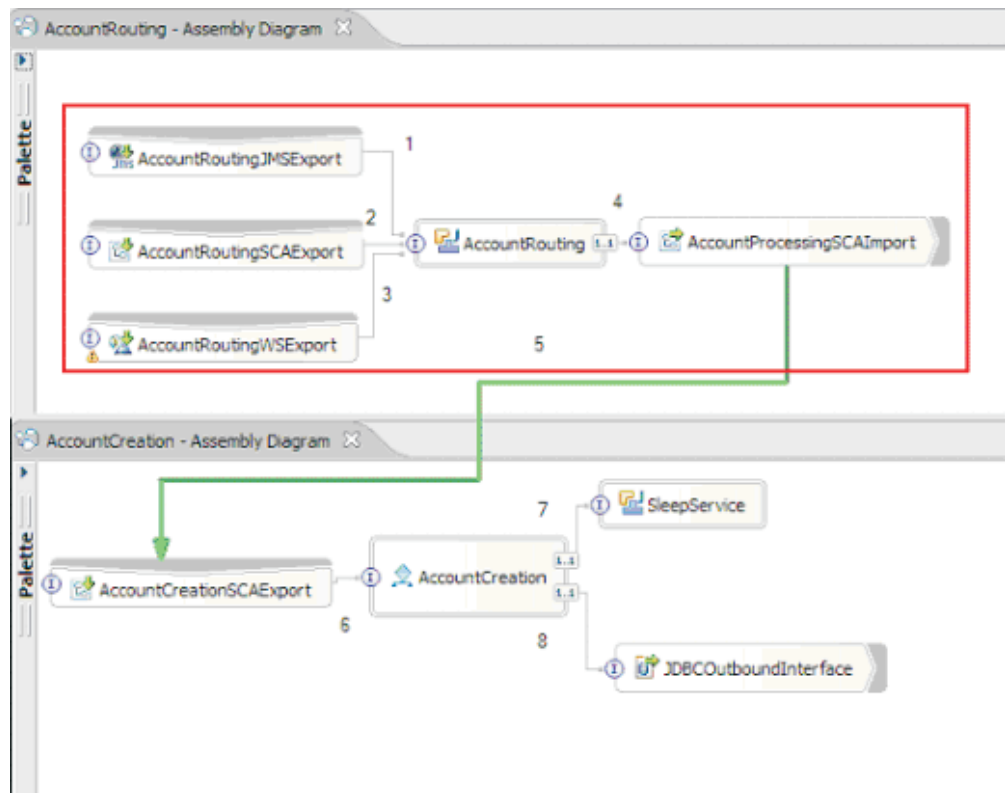
The connectivity group will not contain stateful component types like long-running business processes and Business State Machines. These connectivity groups provide encapsulation and isolation of the specific endpoint's integration requirements. Commonly, WebSphere ESB mediation modules are used for this purpose as they represent convenient ways implement "infrastructure" related tasks.

The concept of connectivity groups also provide a convenient way to quiesce the system in case there is a need for recovery. Because the connectivity group module is stateless, the module can be temporarily stopped thus cutting off the inbound flow of new events while the system finishes processing the events it has.

Note: If you want to stop the flow of inbound events, then the connectivity modules **should not** support inbound and outbound in the same module (even though the same EIS system may have both inbound and outbound). If inbound and outbound support are in the same module, then the outbound is turned off with the inbound. This may cause internal work to stop from completing. Consider separating inbound and outbound in this case.

When the system is recovered and able to process new work, these modules can be restarted.

The module that is outlined in the following screen capture is considered part of a connectivity group.



Connectivity groups can be used for input from an external source or an existing system such as SAP or CICS®. Or for new work from a web browser-based clients.

Application design considerations for exceptions and faults:

You need to consider your application design so that it can take advantage of the error handling and fault processing capabilities in WebSphere ESB.

In order to create a comprehensive error handling strategy, solution architects need to understand how WebSphere Process Server and WebSphere ESB represent declared and undeclared exceptions.

The SCA programming model provides two types of exceptions:

- Service Business Exceptions

Service Business Exceptions are checked exceptions declared in a business method's function signature (WSDL faults or Java throws). Service Business Exceptions identify error conditions that are anticipated by the application or service. These exceptions are sometimes referred to as "checked exceptions"

An example is an `InvalidSymbolException` for a stock quote service. Such exceptions are wrapped by `ServiceBusinessException` and passed back to the client.

- Service Runtime Exceptions

Also known as "system exceptions" service runtime exceptions are not declared in the method signature. In general, they represent error conditions that are not anticipated by the application, such as a `NullPointerException` in a Java Component.

These exceptions are wrapped by `ServiceRuntimeException` and passed back to the client, which can interrogate the `ServiceRuntimeException` to determine the cause.

Note: When working at the SCA level these exceptions are sometimes referred to as faults. However, when using Java code they are usually referred to as exceptions.

When a `ServiceRuntimeException` is thrown from a component, the current transaction will be rolled back.

Service Business Exception handling:

Service Business Exceptions represent known and declared exceptions anticipated by the application or service.

Service Business Exceptions are defined on the service interface.

Component developers should take care to declare the possible exceptions that may be thrown, so that the consuming service can handle them. For example, a business fault to a banking application would include “Invalid Account Number”, or “Insufficient Funds” as *business exceptions*. So the application that calls the service needs to include logic to handle a situation where they have passed in an invalid account number, or where they tried to transfer \$100 but there was only \$50 in the account. These are the types of business errors that a calling application is designed to handle. The WebSphere ESB business exceptions are returned to the client to catch and handle appropriately.

When handling business service exceptions, service consumers should implement the client such that it will perform one of the following actions for a declared business exception:

1. Catch the exception and create the appropriate Service Business Exception for the calling application.
This could mean including the original exception in the new exception (wrapping it). This is most often done when the calling module does not have the same Business Exceptions as the service that it is calling. Here is an example of the flow catching an exception and creating a Service Business Exception for the calling application:
 - a. Module A has SBE “MoneyTransferFailed”
 - b. Module B has SBE “InsufficientFunds”
 - c. Module A calls Module B and gets “InsufficientFunds” exception
 - d. Module A must create a new exception “MoneyTransferFailed”, which may have a place where a string defining the original error of insufficient funds can be included.
2. Catch the exception and perform alternate logic.

Service Runtime Exception handling:

Service Runtime Exceptions are undeclared exceptions. In general, they represent error conditions that are not anticipated by the application.

Service Runtime Exceptions are used to signal an unexpected condition in the runtime.

Component developers can handle Service Runtime Exceptions in the following ways:

1. Catch them and perform some alternative logic.
For example, if one partner is not able to service a request perhaps another one might.

2. Catch the exception and "re-throw" it to your client.
3. Remap the exception to a business exception.

For example, a timeout for a partner may result in a business exception that indicates most of the request was processed but there was one piece of the request that was not completed and should be retried later or tried with different parameters.

If an exception is not caught, the exception is passed on to the component that called the current component. This call chain continues back to the original caller in the chain. For example, Module A calls Module B and Module B calls Module C and then Module C throws an exception, Module B might or might not catch the exception. If Module B does not catch the exception, then the exception travels back to Module A.

When a `ServiceRuntimeException` is thrown from a component, the current transaction will be rolled back. This type of exception processing is repeated for all components in the chain. For example, if a `ServiceRuntimeException` is thrown from Module C, that transaction will be marked for rollback. Then the exception is thrown to Module B, where if it is not caught and another transaction is present, that transaction also will be rolled back. Component developers can use quality of service (QoS) qualifiers to control whether invocations occur in the current transaction or a new transaction. So, if Module A calls Module B and Module B is part of a new transaction, then Module A can "catch" a `ServiceRuntimeException` from Module B and continue processing, without Module A's transaction rolling back.

Note: Because runtime exceptions are not declared as part of the interface, component developers should attempt to resolve the exception and thus prevent a runtime exception from inadvertently being propagated to the client if the client is a user interface.

You should be aware that the contents of the rolled back transaction can vary, depending on the nature of the transaction. For example, long-running BPEL processes can be segmented into many smaller transactions. Asynchronous request and response calls are broken out of a transaction automatically (otherwise the calling application might have to wait a long time for the response).

In instances where a transaction is broken into multiple asynchronous calls (as opposed to one large transaction), the initial work for the transaction would rollback at the occurrence of a `ServiceRuntimeException`. However, the response from the asynchronous call is sent from a different transaction, and because the response from the asynchronous call would have no place to go, an event is created in the Failed Event Manager (FEM).

The following list is of 4 current subclasses of `ServiceRuntimeException`:

1. `ServiceExpirationRuntimeException`
This exception is used to indicate that an asynchronous SCA message has expired. Expiration times can be set using the `RequestExpiration` qualifier on a service reference.
2. `ServiceTimeoutRuntimeException`
This exception is used to indicate that the response to an asynchronous request was not received within the configured period of time. Expiration times can be set using the `ResponseExpiration` qualifier on a service reference.
3. `ServiceUnavailableException`

This exception is used to indicate that there was an exception thrown while invoking an external service via an import.

4. `ServiceUnwiredReferenceRuntimeException`

This exception is used to indicate that the service reference on the component is not wired correctly.

Preparing to install and configure the software

Before preparing to install and configure the software, create a plan for the deployment environment that you want to create.

Use the information listed in the following table to prepare for installing and configuring WebSphere ESB.

Table 36. Preparing for installation and configuration

| Task | Where to find information | Result after completing the task |
|---|---|--|
| Review hardware and software requirements | For hardware and software requirements, visit http://www-01.ibm.com/software/integration/wsesb/sysreqs/ | You understand the system requirements necessary to support your WebSphere ESB installation. |
| Prepare your operating system | <div><div>AIX</div><div><div>HP-UX</div>Preparing HP-UX systems for installation</div><div><div>Linux</div>Preparing Linux systems for installation</div><div><div>Solaris</div></div><div><div>Windows</div>Preparing Windows systems for installation</div></div> | You have prepared the operating system of each workstation to be used. |

Table 36. Preparing for installation and configuration (continued)

| Task | Where to find information | Result after completing the task |
|---|---|---|
| Make sure you have installed your database management system. | Consult your database documentation for information about installing and administering your database management system. | <p>Your database management system is installed.</p> <p>WebSphere ESB embeds the DB2 Express database. If you want to use DB2 Express as your database, you can select it as a feature from the installer and it is installed and configured automatically.</p> <p>Note: If you already have a version of DB2 installed and you want to install DB2 Express, you must uninstall DB2 before running the WebSphere ESB installer. If the installer detects a version of DB2 installed and you have selected to install DB2 Express from the installer, you will receive a warning message and will not be able to install DB2 Express.</p> <p>Important: Linux If you are installing DB2 Express as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. See Kernel parameter requirements (Linux) for a list of the kernel requirements. You can locate the current values by parsing the output of the <code>ipcs -l</code> command.</p> |

Preparing operating systems for product installation

Before you can install IBM Business Process Manager, you must prepare your operating system. The configuration depends on the type of operating system you are using.

Before you begin

Before preparing the installation environment, complete the following tasks:

- Disable the firewall if you have a firewall running on the system where you plan to install IBM Business Process Manager.
- Ensure that your user login provides access to your DB2 or Oracle database commands.
- Complete additional tasks specific to your operating system.

Attention: While installing IBM Business Process Manager 7.5.0 on **Windows Server 2003** or **Windows 7 (English)** and switching the locale to Czech in the IBM Process PortalPreference -> Interface language the text displays with corrupt characters on several panels of the IBM Process Designer. The text displays with corrupt characters even if IBM Process Designer is started with the Czech locale.

Tip: To resolve this issue change system settings as follows.

- In **Windows Server 2003**: Regional and Language Options -> Advanced -> Language for non-Unicode programs -> set to 'Czech'

- In **Windows 7**: Regional and Language Options -> Administrative Tab -> Change system locale... (under 'Language for non-Unicode programs') -> Select 'Czech'

Preparing AIX systems for installation

Before you can install WebSphere ESB, you must prepare your AIX® operating system.

Before you begin

Because WebSphere Application Server is a prerequisite of WebSphere ESB, you must complete the required preparation steps in the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

About this task

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system.

Refer to the following technote for additional preparation information for configuring Installation manager to run on 64-bit AIX systems:
<http://www-304.ibm.com/support/docview.wss?uid=swg21330190&wv=1> .

Procedure

Complete the following steps on your AIX system before installing WebSphere ESB:

1. Increase the maximum number of open files. The default setting is usually not enough. You can check your current maximum number of open files by using `ulimit -n`. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The `ulimit` requirement is dynamically calculated at installation time and might need to be larger based on the options you select.

Before installing, run the following command:

```
ulimit -n 8800
```

Alternatively, you can use the following steps to edit the resource limits file:

- a. Open `/etc/security/limits`.
 - b. Edit or add the **default** section and include this line:

```
nofiles = 8800
```
 - c. Save and close the file.
 - d. Log off from the operating system and log in again.
2. Set the **umask** value to 022 using the following command:

```
umask 022
```
 3. Ensure that you have Mozilla Firefox installed at version 3.5.x.x or higher.
 4. Before starting the data movement service, increase the number of processes configured in the AIX operating system to avoid a connection reset error. You can increase the number of processing using a command, or using the AIX interface.
 - Run the command:

```
chgdev -l sys0 -a maxuproc='256'
```

- In the AIX interface, enter **smitty**, then select **System Environments > Change / Show Characteristics of Operating System > Number of processes allowed per user(Num.)**.

5. Complete the steps to Tune AIX systems.

Preparing HP-UX systems for installation

Before you can install WebSphere ESB, you must prepare your HP-UX operating system.

Before you begin

Because WebSphere Application Server is a prerequisite of WebSphere ESB, you must complete the required preparation steps in the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

About this task

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system.

Procedure

Complete the following steps on your HP-UX system before installing WebSphere ESB:

1. Increase the maximum number of open files. The default setting is usually not enough. You can check your current maximum number of open files by using `ulimit -n`. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The `ulimit` requirement is dynamically calculated at installation time and might need to be larger based on the options you select.

Before installing, run the following command:

```
ulimit -n 8800
```

Alternatively, you can use the following steps to edit the resource limits file:

- a. Open `/etc/security/limits`.
- b. Edit or add the **default** section and include this line:

```
nofiles = 8800
```

- c. Save and close the file.
- d. Log off from the operating system and log in again.

2. Set the **umask** value to 022 using the following command:

```
umask 022
```

3. Complete the steps to Tune HP-UX systems.

Preparing Linux systems for installation

Before you can install WebSphere ESB, you must prepare your Linux operating system.

Before you begin

Because WebSphere Application Server is a prerequisite of WebSphere ESB, you must complete all the required preparation steps in the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

Ensure that you have Mozilla Firefox installed at version 3.5.x.x or higher.

About this task

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system. To install Installation Manager on Red Hat Enterprise Linux 6.0 (64-bit), see [Unable to install Installation Manager on RHEL 6.0 \(64-bit\)](#).

If you are planning to install WebSphere ESB using DB2 Express with Red Hat Enterprise Linux 6 as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the `ipcs -l` command.

To change the values:

1. Add the following lines, in the below order, to the `/etc/sysctl.conf` file:

```
kernel.shmmni=4096
kernel.shmmax=4294967296
kernel.shmall=8388608
#kernel.sem=<SEMMS><SEMMNS><SEMOPM><SEMMNI>
kernel.sem=250 256000 32 4096
kernel.msgmni=16384
kernel.msgmax=65536
kernel.msgmnb=65536
```

2. Add the following lines to the end of `/etc/security/limits.conf`:

```
# - stack - max stack size (KB)
* soft stack 32768
* hard stack 32768
# - nofile - max number of open files
* soft nofile 65536
* hard nofile 65536
# - nproc - max number of processes
* soft nproc 16384
* hard nproc 16384
```

3. Reboot your system.

Procedure

Complete the following steps on your Linux system before installing WebSphere ESB:

1. Increase the maximum number of open files. The default setting is usually not enough. You can check your current maximum number of open files by using `ulimit -n`. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The `ulimit` requirement is dynamically calculated at installation time and might need to be larger based on the options you select.
 - a. Open `/etc/security/limits.conf`.

- b. Locate the `nofile` parameter and increase the value. If a line containing the `nofile` parameter does not exist, add the following lines to the file:
 - * `hard nofile 8800`
 - * `soft nofile 8800`
- c. Save and close the file.
- d. Log off and log in again.

For more information about this setting, run `man limits.conf` or see the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

2. Install the following packages for your operating system:

| Option | Description |
|---|---|
| Red Hat Enterprise Linux 5 | compat-libstdc++-33-3.2.3-61 compat-db-4.2.52-5.1 libXp-1.0.0-8 rpm-build-4.4.2-37.el5 64-bit kernel only: compat-libstdc++-296-2.96-138 |
| Red Hat Enterprise Linux 6 | ksh-version.rpm Korn shell See the detailed instructions and list of packages in Unable to install Installation Manager on RHEL 6.0 (64-bit) |
| SUSE Linux Enterprise Server 9.0 | XFree86-libs-32bit-9 glibc-32bit-9 glib-32bit-9 gtk-32bit-9 |

You can also install a later release of any of these packages if there are new packages as errata. If you have additional packages that are specific to your hardware, install them.

You can use single-line commands to install dependencies (all required packages). The following commands are examples using the default package managers on supported Linux distributions.

- **Red Hat Enterprise Linux 5 (32-bit):**
`yum install compat-libstdc++-33 compat-db libXp rpm-build RHEL 5.x`
- **Red Hat Enterprise Linux 5 (64-bit):**
`yum install compat-libstdc++-33 compat-db libXp rpm-build compat-libstdc++-296`
- **SUSE Linux:**
`zypper install XFree86-libs-32bit-9 glibc-32bit-9 glib-32bit-9 gtk-32bit-9`

3. Set the **umask** value to 022 using the following command:
`umask 022`
4. On Red Hat Enterprise Linux 5 systems, disable SELinux, or set it to a permissive mode.
5. Restart the computer.
6. Complete the steps to Tune Linux systems.

Preparing Solaris systems for installation

Before you can install WebSphere ESB, you must prepare your Solaris operating system.

Before you begin

Because WebSphere Application Server is a prerequisite of WebSphere ESB, you must complete the required preparation steps in the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

About this task

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system.

Refer to the following technote for additional preparation information for configuring Installation manager to run on 64-bit AIX systems:

<http://www-01.ibm.com/support/docview.wss?uid=swg24027719>

Procedure

Complete the following steps on your Solaris systems before installing WebSphere ESB:

1. Increase the maximum number of open files. The default setting is usually not enough. You can check your current maximum number of open files by using `ulimit -n`. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The `ulimit` requirement is dynamically calculated at installation time and might need to be larger based on the options you select.

Before installing, run the following command:

```
ulimit -Hn 8800
```

Alternatively, you can use the following steps to edit the resource limits file:

- a. Open `/etc/system`
 - b. Add the following line to the end of the file:

```
set rlim_fd_max=8800
```
 - c. Save and close the file.
 - d. Log off from the operating system and log in again.
2. Set the `umask` value to `022` using the following command:

```
umask 022
```
 3. Complete the steps to Tune Solaris systems.

Preparing Windows systems for installation

Before you can install WebSphere ESB, you must prepare your Windows operating system.

Before you begin

If you are planning to use DB2 Express with your WebSphere ESB installation, the user account must belong to the Administrators group on the machine where you will perform the installation.

About this task

Because WebSphere Application Server is a prerequisite product for WebSphere ESB, you must complete all of the preparation tasks for WebSphere Application Server before installing WebSphere ESB.

Procedure

Complete the following steps on your Windows system before installing WebSphere ESB:

1. Complete the steps in the Preparing Windows systems for installation topic in the WebSphere Application Server information center.
2. Complete the steps to Tune Windows systems.

Creating the Common database manually before product installation

Use these instructions if you decide to create the Common database manually.

About this task

WebSphere ESB provides default scripts that you can use to create the Common database manually. You might want to create the database manually in the following situations:

- If your organization requires that the database be created by a user with DBA privileges, that user must create the Common database before creating or augmenting profiles.
- If you intend to create or augment profiles during product installation, a user with DBA privileges must create the Common database before you install WebSphere ESB.

Procedure

1. Go to the directory that contains the database creation scripts. The scripts are located both on the product media and in a directory after product installation. By default, the scripts are located in the following directories:
 - Location on the product media:
 - **Linux** **UNIX** `<media_root>/dbscripts` or `<extract_directory>/dbscripts`
 - **Windows** `<media_root>\dbscripts` or `<extract_directory>\dbscripts`
 - Location after installation:
 - **Linux** **UNIX** `install_root/dbscripts`
 - **Windows** `install_root\dbscripts`
2. Open the directory containing the Common database scripts for your database product. The default location depends on the platform:
 - **Linux** **UNIX** `.../CommonDB/db_type`
 - **Windows** `...\CommonDB\db_type`

The variable `db_type` represents the supported database type. Refer to Table 37 on page 75 to locate your database type and directory name.

Applicable database types and their directory names are as follows:

Table 37. Applicable database types and their directory names

| Database type | Directory name | Corresponding subtopic |
|---|----------------|---|
| DB2 Universal Database™ (for all operating systems except z/OS) | DB2 | "Creating the DB2 database" |
| DB2 for z/OS | DB2zOS | "Creating the DB2 database for z/OS" on page 76 |
| Oracle | Oracle | "Creating the Oracle database" on page 77 |
| Microsoft SQL Server | SQLServer | "Creating the Microsoft SQL Server database" on page 79 |

- Click the corresponding subtopic link in Table 37 to proceed with creating the common database manually.

Creating the DB2 database

In order to create a Common database manually, you need to edit and run the scripts that come with WebSphere ESB. This topic tells you how to edit and run scripts associated with the DB2 database.

About this task

Before you can run scripts to create a DB2 database manually, you need to customize them for WebSphere ESB. WebSphere ESB comes with following scripts:

Table 38. DB2 scripts for WebSphere ESB

| |
|--------------------------------------|
| configCommonDB.bat |
| configCommonDB.sh |
| createDBTables.bat |
| createDBTables.sh |
| createTable_CommonDB.sql |
| createTable_lockmanager.sql |
| createTable_Recovery.sql |
| createTable_EsbLoggerMediation.sql |
| createTable_governancerepository.sql |
| insertTable_CommonDB.sql |
| createTable_Relationship.sql |
| createTable_RelationshipService.sql |

Procedure

- Make sure that you are using a user ID with sufficient authority to update the database schema.
- Locate the directory where the database scripts are located:
 - Linux** **UNIX** `<media_root>/dbscripts/CommonDB` or `<extract_directory>/dbscripts/CommonDB`
 - Windows** `<media_root>\dbscripts\CommonDB` or `<extract_directory>\dbscripts\CommonDB`
- Locate the configCommonDB.bat or configCommonDB.sh file and perform the following subtasks:

- a. Replace the *DB_NAME* variable with the database name, for example WPRCSDB.
- b. Replace the *USER_NAME* variable with the database user name, for example db2admin.

You must pass the **createDB** parameter to the configCommonDB script if you want to create a new local database; otherwise an existing database will be used. For example:

configCommonDB.sh createDB - create tables in a new database

configCommonDB.sh - create tables using an existing database

Important: You need to have *SEC0FR authority on the IBM i system before you can run these scripts.

4. Locate the createDatabase_CommonDB.sql file and perform the following subtask.
 - a. Replace the *DB_NAME* variable with the database name, for example WPRCSDB.
5. Run the configCommonDB.bat or configCommonDB.sh script. This in turn will run the createDBTables.bat or createDBTables.sh script to create the necessary schema and tables for the Common database.
6. If there are any errors, or failure is indicated in your database client output, fix the reported errors and try again.

Results

The DB2 database is created.

Creating the DB2 database for z/OS

In order to create a Common database manually, you need to edit and run the scripts that come with WebSphere ESB. This topic tells you how to edit and run scripts associated with the DB2 for z/OS database.

About this task

Before you can run scripts to create a DB2 for z/OS database manually, you need to customize them for WebSphere ESB. WebSphere ESB comes with following scripts:

Table 39. DB2 for z/OS scripts for WebSphere ESB

| |
|--------------------------------------|
| createTable_CommonDB.sql |
| createTable_EsbLoggerMediation.sql |
| createTable_Recovery.sql |
| createTable_RelationshipService.sql |
| createTable_governancerepository.sql |
| createTable_lockmanager.sql |
| insertTable_CommonDB.sql |

Procedure

1. Make sure that you are using a user ID with sufficient authority to update the database schema.
2. Locate the directory where the database scripts are located:

- **Linux** **UNIX** `<media_root>/dbscripts/CommonDB` or `<extract_directory>/dbscripts/CommonDB`
 - **Windows** `<media_root>\dbscripts\CommonDB` or `<extract_directory>\dbscripts\CommonDB`
3. Replace the following variables in the DB2 for z/OS scripts, which are located in the DB2zOS directory, with your database-specific information: @DB_NAME@, @STOGRP@, and @SCHEMA@.
 4. Run the DB2 for z/OS scripts, which are listed in Table 39 on page 76. For information about how to run a .sql script with your database, refer to the documentation for your database product.
 5. If there are any errors, or failure is indicated in your database client output, fix the reported errors and try again.

Results

The DB2 for z/OS database is created.

Example

The createTable_lockmanager.sql script is missing from under the dbscripts/CommonDB/DB2zOS folder in the WebSphere Enterprise Service Bus V7.0 CD image.

You can use the scripts from the CD image directly to set up their databases and not necessarily wait to install the entire product and, or create profiles to get a hold of these scripts.

If you use the scripts from the CD image to create the Common Database, you will miss the createTable_lockmanager.sql script which may cause runtime issues with response to these tables.

However, this file does show up after WebSphere ESB is installed, under the <INSTALL>/dbscripts/CommonDB/DB2zOS and also after a profile is created (under the profiles/<profiles>/dbscripts folder)

To fix this, install the product and then use the scripts from under the <INSTALL>/dbscripts/CommonDB/DB2zOS location.

Creating the Oracle database

In order to create a Common database manually, you need to edit and run the scripts that come with the WebSphere ESB. This topic tells you how to edit and run scripts associated with the Oracle database.

About this task

Before you can run scripts to create an Oracle database manually, you need to customize them for WebSphere ESB. WebSphere ESB comes with following scripts:

Table 40. Oracle scripts for WebSphere ESB

| |
|-----------------------------|
| configCommonDB.bat |
| configCommonDB.sh |
| createDatabase_commonDB.sql |
| createTable_commonDB.sql |

Table 40. Oracle scripts for WebSphere ESB (continued)

| |
|--|
| createTable_EsbLoggerMediation.sql |
| createTable_governancerepository.sql |
| createTable_lockmanager.sql |
| createTable_Recovery.sql |
| createTable_RelationshipMetadataTable.sql |
| insertTable_CommonDB.sql |
| createTable_RelationshipViewMetaaTable.sql |

Procedure

1. Make sure that you are using a user ID with sufficient authority to update the database schema.
2. Locate the directory where the database scripts are located:
 - **Linux** **UNIX** `media_root/dbscripts/CommonDB/oracle` or `<extract_directory>/dbscripts/CommonDB/oracle`
 - **Windows** `media_root\dbscripts\CommonDB\oracle` or `<extract_directory>\dbscripts\CommonDB\oracle`
3. Locate the configCommonDB.bat or configCommonDB.sh file and perform following subtasks:
 - a. Replace the `DB_NAME` variable with the Oracle Database name [SID], for example ORCL.
 - b. Replace the `DB_USER` variable with Oracle user, for example orcCOMM.
4. Locate the createSchema_CommonDB.sql file which is a template used to create required schemas. To create a database schema:
 - a. Replace the `DB_USER` variable with the database schema name. For example, orcCOMM.
 - b. Replace the `dbCommonPassword` variable with the database schema password. For example, youNameIt. If not changed, you will be requested to enter a password for the `DB_USER`.
 - c. Repeat the above steps for each additional schema.
 - d. Required: Run the createSchema_CommonDB.sql script.

The following components require a schema. These schemas will be generated automatically if not passed during profile creation. The default schemas are:

Note: The *SID* value shown below is the first 3 characters of the Oracle Database name. For example, orcCOMM.

Table 41. Default schemas

| Component | Default value |
|----------------|----------------|
| CommonDB | <i>SIDCOMM</i> |
| Business Space | IBMBUSSP |
| SCA.SYSTEM ME | <i>SIDSS00</i> |
| SCA.APP ME | <i>SIDSA00</i> |
| CEI ME | <i>SIDCM00</i> |
| BPC ME | <i>SIDBM00</i> |
| CEI | <i>SIDCEID</i> |

For the above parameters, the value of the password depends on how you configure the profile. The Value can be a dbPassword or the value that is used while running the manageprofiles command-line utility. To run these scripts you must have SYSDBA privileges.

5. Copy all of the scripts from the *extract_directory*\dbscripts\CommonDB directory to the Oracle workstation and run the configCommonDB.bat or configCommonDB.sh script.

Note: Confirm that the database schema name that was specified in step 4 on page 78 above, for example, orcCOMM, is created before running this script because it uses the database schema name to connect the database for creating tables.

6. If there are any errors, or failure is indicated in your database client output, fix the reported errors and try again.

Results

The Oracle database is created.

Creating the Microsoft SQL Server database

In order to create a Common database manually, you need to edit and run the scripts that come with WebSphere ESB. This topic tells you how to edit and run scripts associated with the Microsoft SQL Server database.

About this task

Before you can run scripts to create a Microsoft SQL Server database manually, you need to customize them for WebSphere ESB. WebSphere ESB comes with following scripts:

Table 42. Microsoft SQL Server scripts for WebSphere ESB

| |
|-------------------------------------|
| createDatabase_CommonDB.sql |
| createTable_RelationshipService.sql |
| dropTable_AppScheduler.sql |
| createTable_CommonDB.sql |
| createTable_lockmanager.sql |
| createTable_Recovery.sql |
| createTable_EsbLoggerMediation.sql |
| insertTable_CommonDB.sql |
| configCommonDB.bat |

Procedure

1. Make sure that you are using a user ID with sufficient authority to update the database schema.
2. Locate the directory where the database scripts are located:
 - Linux UNIX <media_root>/dbscripts/CommonDB or <extract_directory>/dbscripts/CommonDB
 - Windows <media_root>\dbscripts\CommonDB or <extract_directory>\dbscripts\CommonDB

3. Locate the configCommonDB.bat or configCommonDB.sh script and perform the following subtasks:
 - a. Replace the *DB_NAME* variable with the database name, for example WPRCSDB.
 - b. Replace the *DB_USER* variable with the database user name, for example sqluser.
 - c. Replace the *DB_HOSTNAME* variable with the SQL host name, for example me.usca.ibm.com.
4. Run the configCommonDB.bat or configCommonDB.sh script which was modified in step 3.
5. If there are any errors, or failure is indicated in your database client output, fix the reported errors and try again.

Results

The Microsoft SQL Server database is created.

Installing WebSphere Enterprise Service Bus

Installing WebSphere Enterprise Service Bus involves acquiring the software and then installing the software files: prerequisite software, the database product to be used by WebSphere Enterprise Service Bus, and the WebSphere Enterprise Service Bus software. You can either install the software interactively from the launchpad program or silently by running Installation Manager in silent installation mode. You can also choose to install WebSphere ESB on top of an existing WebSphere Application Server.

Installing the software interactively

Install WebSphere ESB by specifying your choices interactively with the installation graphical user interfaces. Optionally, you can also create a stand-alone server profile to evaluate the product or support application development.

Before you begin

If you are planning to install WebSphere ESB using DB2 Express with Red Hat Enterprise Linux 6 as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the `ipcs -l` command.

Windows To install or run WebSphere Enterprise Service Bus on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.

About this task

This task describes installation of the main product software, to be used to create your chosen deployment environment. For information about installing other software provided with the product (including the message service clients, the product Help System and documentation, or additional software) see other topics.

Procedure

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage®.
 - If you are installing from the product DVD, perform the following steps:
 - a. Insert the product disk labeled WebSphere Enterprise Service Bus into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to the next step. If autorun is not enabled on your workstation, enter one of the following commands to start the launchpad manually:
 - **Linux** **UNIX** `mount_point/launchpad.sh`
 - **Windows** (from a command line) `DVD_root\launchpad.exe`
 - If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter one of the following commands to start the launchpad:
 - **Linux** **UNIX** `extract_directory/launchpad.sh`
 - **Windows** (from a command line) `extract_directory\launchpad.exe`
2. Optional: If you are not an administrative user, or if you want to install to your own user name without administrative privileges, clear the **Install as administrative user** check box. If you are in the Administrator group on Windows, or if you are a root user on Linux or UNIX systems, you can install as an administrative user.
3. Click **Install** to start the IBM Installation Manager.

Important: If you are on a 64-bit system, you might receive the following message:

Your operating system failed the launchpad prerequisites check. The following 32-bit GTK library for running IBM Installation Manager is not available in underlying OS: `list_of_missing_files`. Please install the 32-bit GTK library and restart your installation.

If you see this message, your server does not have the 32-bit version of the GTK library installed, or the library is an incorrect version. Update your server with the correct version of the 32-bit GTK library, using the DVD or official web site of your operating system, before you continue the installation.

4. On the Select packages to install page, select the packages to be installed. When you install WebSphere Enterprise Service Bus, the required WebSphere Application Server Network Deployment, Feature Pack for XML, and Feature Pack for SCA are automatically installed.
 - a. Optional: Select IBM DB2 Express to install and use an embedded DB2 Express database.

Restriction: If you intend creating the profile for a stand-alone development environment (qesb) while installing the software, you must select the IBM Express check box.

Click **Next** to continue.

5. The installation program performs prerequisite checks.

Attention: If you receive any of the following error messages during the prerequisite check, address the product incompatibility issues, click **Back**, fix the problem and click **Next** to continue:

- If you selected to install DB2 Express and DB2 is already installed on this system, you will receive the following error message:
DB2 is already installed on this system. Either uninstall existing DB2 or deselect DB2 Express from being installed.
- If you select to install WebSphere ESB V7.5 to the same WebSphere Application Server location as any other WebSphere ESB V7.5 or equivalent, earlier release products, you receive the following error message:
IBM WebSphere Enterprise Service Bus V7.5 can not coexist with the following offerings:

Installation Manager lists all incompatible offerings for your reference.

Important:

If you receive the following warning message during the prerequisite checking, use the platform-specific steps below to increase the `ulimit` number.

Current system has detected a lower level of ulimit than the recommended value of recommended_value. Please increase the ulimit number to minimum value of recommended_value and re-start the installation. Shutdown your installer. If you are a root user open a command prompt and issue ulimit -n recommended_value and then restart the installer. If you are a non-root user, work with your system administrator to increase your ulimit -n recommended_value and then restart the installer.

The required value is calculated based on the version of WebSphere Application Server, the feature packs, and the configuration that you are installing.

- Set the maximum number of open files using the following steps: Linux
 - Open `/etc/security/limits.conf`.
 - Locate the `nofile` parameter and increase the value. If a line containing the `nofile` parameter does not exist, add the following lines to the file:





```
* hard nofile recommended_value
* soft nofile recommended_value
```
 - Save and close the file.
 - Log off and log in again.
 - Restart the computer.
 - Restart the installer.
- On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms of the license agreements** then click **Next** to continue.
 - On the Location page, specify the directory where you want to install WebSphere ESB. In the **Installation Directory** field, specify the full path to the directory.
Click **Next** to continue.
 - On the Features page, select the package features that you want to install.
 - Optional: To see the dependency relationships between features, select **Show Dependencies**.
 - Optional: Click a feature to view its brief description under **Details**.
 - Optional: If you want to create the profile for a stand-alone development environment (qesb) while installing the software, expand **IBM WebSphere Enterprise Service Bus 7.5.0.0**, then select the option **Stand-alone development WebSphere Enterprise Service Bus profile (qesb)**.

Tip: This stand-alone environment is intended to help you evaluate the product or to support application development. Creating this profile requires you to supply your administrator security ID and password credentials.

- d. Optional: If you want to install the sample applications, expand **IBM WebSphere Enterprise Service Bus 7.5.0.0**, then select the option **Sample applications**. This option installs the sample applications for both WebSphere ESB and WebSphere Application Server Network Deployment. Sample applications include both source code files and integrated enterprise applications that demonstrate some of the latest Java Platform, Enterprise Edition (Java EE) and WebSphere technologies.

Tip: For better performance in a production environment, do not install the Sample Applications.

Click **Next** to continue.

9. On the Common Configurations page, specify the credentials requested.
 - a. Optional: Profile Configuration If you have selected to create the stand-alone development environment (qesb), specify the user name and password to be used for actions on that profile.
 - b. Optional: DB2 Credentials If you selected to install and use an embedded DB2 Express database, specify the DB2 administrative user name and password. The default values are:
 -  Instance user name and password: bpminst and bpminst1
 -  Fenced user name and password: bpmfenc and bpmfenc1
 -  Administration server (DAS) user name and password: bpmadmin and bpmadmin1
 -  Administrative user name and password: bpmadmin and bpmadmin1

Restriction: User names must not contain NL strings.

Click **Next** to continue.

10. On the Summary page, review your choices before installing the WebSphere ESB package. If you want to change any choices that you made on previous pages, click **Back** then make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.
11. When the installation process is complete, a message confirms the success of the process. If you chose to create a stand-alone development profile during installation and it failed or did not fully succeed, you see an error message informing you of the failure and giving you the location of the profile creation error log at *install_root/logs/manageprofiles/profilename_create.log*. You must resolve the profile creation problem and create a profile using the Profile Management Tool or the **manageprofiles** command.
 - a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.
 - b. Select **Profile Management Tool** if you want to launch the Profile Management Tool when you finish or select **None** to complete the installation.
 - c. Click **Finish** to close the Installation Manager.

What to do next

If you created the stand-alone development environment (qesb), you can start the First Steps console to verify your installation, start or stop the stand-alone server, access the administrative console, and perform other actions.

Otherwise, you can use the Profile Management Tool or manageprofiles command-line utility to create profiles for stand-alone servers, deployment managers, and custom (managed) nodes. For a network deployment environment, you can then go on to create the deployment environment that you need.

Installing on an existing installation of WebSphere Application Server

You can install WebSphere Enterprise Service Bus (WebSphere ESB) on an existing installation of WebSphere Application Server.

Procedure

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage.
 - If you are installing from the product DVD, insert the product disk labeled WebSphere Enterprise Service Bus into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to step 2. If autorun is not enabled on your workstation, enter one of the following commands to start the launchpad manually:
 - **Linux** **UNIX** `mount_point/launchpad.sh`
 - **Windows** (from a command line) `DVD_root\launchpad.exe`
 - If you are installing from images downloaded from Passport Advantage, perform the following steps:
 - a. Go to the directory into which you extracted the images.
 - b. Enter one of the following commands to start the launchpad:
 - **Linux** **UNIX** `extract_directory/launchpad.sh`
 - **Windows** (from a command line) `extract_directory\launchpad.exe`

Important: If you are on a 64-bit system, you might receive the following message:

Your operating system failed the launchpad prerequisites check. The following 32-bit GTK Library for running IBM Installation Manager is not available in underlying OS: list_of_missing_files. Please install the 32-bit GTK Library and restart your installation.

If you see this message, your server does not have the 32-bit version of the GTK library installed, or the library is an incorrect version. Update your server with the correct version of the 32-bit GTK library, using the DVD or official web site of your operating system, before you continue the installation.

2. Click **Installation on an existing WebSphere Application Server**.
3. Select **Install as administrative user** to install as an administrative user. If you are in the Administrator group on Windows, or if you are a root user on Linux or UNIX systems, you can install as an administrative user. If you are not an administrative user, or if you want to install to your own user name without administrative privileges, clear this check box.
4. Click **Import or Update** to import or update WebSphere Application Server and associated feature packs.

If WebSphere Application Server has never been imported before, or if it has been updated with the Update Installer after it was last imported, you must import WebSphere Application Server now. Click Import or Update now, and then, when Installation Manager opens, click **Import**.

If WebSphere Application Server has been imported before, and has not been updated, you can update WebSphere Application Server now. Click **Import or Update** now, and then, when Installation Manager opens, click **Update**. Install available updates for WebSphere Application Server, Feature Pack for XML and Feature Pack for SCA. On the Update Packages page, select **Show all** to display available updates. If you have already installed the Feature Pack for SCA, ensure that the Service Data Objects feature is installed. If not, select the feature on the Feature panel. The Service Data Objects feature requires the Feature Pack for XML.

5. Click **Install** to install WebSphere ESB . When you install WebSphere ESB, the required WebSphere Application Server, Feature Pack for XML, and Feature Pack for SCA are automatically selected for installation. Clear the check boxes beside WebSphere Application Server and the feature packs if they are already installed.

What to do next

Continue to choose options then install the software for WebSphere ESB, as described the IBM Installation Manager panels.

Installing WebSphere Enterprise Service Bus silently

You can install the WebSphere Enterprise Service Bus product package in *silent* installation mode. When you install in silent mode, the user interface is not available.

Installing WebSphere Enterprise Service Bus silently using the command line

You can install WebSphere Enterprise Service Bus using the command line.

Before you begin

Before you install WebSphere Enterprise Service Bus, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

If you are planning to install WebSphere ESB using DB2 Express with Red Hat Enterprise Linux 6 as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the `ipcs -l` command.

If you receive the following warning message during the prerequisite checking, use the platform-specific steps below to increase the `ulimit` number.

Current system has detected a lower level of ulimit than the recommended value of recommended_value. Please increase the ulimit number to minimum value of recommended_value and re-start the installation. Shutdown your installer. If you are a root user open a command prompt and issue ulimit -n recommended_value and then restart the installer. If you are a non-root user, work with your system administrator to increase your ulimit -n recommended_value and then restart the installer.

The required value is calculated based on the version of WebSphere Application Server, the feature packs, and the configuration that you are installing.

1. Set the maximum number of open files using the following steps: Linux
 - a. Open `/etc/security/limits.conf`.
 - b. Locate the `nofile` parameter and increase the value. If a line containing the `nofile` parameter does not exist, add the following lines to the file:

```
* hard nofile recommended_value
* soft nofile recommended_value
```
 - c. Save and close the file.
 - d. Log off and log in again.
2. Restart the computer.
3. Restart the installer.

About this task

If you do not have the prerequisite base products necessary for WebSphere Enterprise Service Bus installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment
- Feature Pack for XML
- Feature Pack for Service Component Architecture (SCA)

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and WebSphere Enterprise Service Bus.

Procedure

To silently install WebSphere Enterprise Service Bus, complete the following steps:

1. Read and accept the license terms before installing. Adding `-acceptLicense` to the command line means that you accept all licenses.
2. Run the following command:

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

Windows

```
extract_directory\imcl install list_of_product_IDs -acceptLicense -installationDirectory location -repositories repository -properties key=value,key=value -showVerboseProgress -log logName.log
```

UNIX

Linux

```
extract_directory\imcl install list_of_product_IDs -acceptLicense -installationDirectory location -repositories repository -properties key=value,key=value -showVerboseProgress -log logName.log
```

where:

- *list_of_product_IDs* is a list of the IDs for the products you want to install, separated by spaces.

Table 43. Product IDs

| Product | Product ID |
|---|--|
| WebSphere ESB | com.ibm.ws.WESB75 |
| WebSphere Application Server Network Deployment | com.ibm.websphere.ND.v70,core.feature,samples,import,productProviders (includes all required features) |
| Feature Pack for Service Component Architecture (SCA) | com.ibm.websphere.SCA.v10 |
| Feature Pack for XML | com.ibm.websphere.XML.v10 |
| Installation Manager | com.ibm.cic.agent,agent_core,agent_jre |
| DB2 for Linux 32-bit | com.ibm.ws.DB2EXP97.linuxia32 |
| DB2 for Linux 64-bit | com.ibm.ws.DB2EXP97.linuxia64 |
| DB2 for Windows 32-bit | com.ibm.ws.DB2EXP97.winia32 |
| DB2 for Windows 64-bit | com.ibm.ws.DB2EXP97.winia64 |

- *location* is the path to the directory where you want to install the products
- *repository* is the path to the repository where you have extracted the files, one of the following directories:
`extract_directory/repository/repos_32bit`
`extract_directory/repository/repos_64bit`
- *key=value* is a list of the keys and values you want to pass to the installation, separated by commas. Do not put spaces between the commas.

Table 44. Keys

| Key | Description |
|----------------------------|---|
| user.select.64bit.image | If you are installing on a 64-bit operating system, add the following line exactly: <code>user.select.64bit.image,,com.ibm.websphere.ND.v70=true</code> The default value is false. |
| user.db2.admin.username | Windows only. User name with authority to access the DB2 database. The default value is bpmadmin. |
| user.db2.admin.password | Windows only. Password for the user name above. The default value is bpmadmin1. |
| user.bpm.admin.username | User name for the administrative console. The default value is admin. This property is needed only if you are creating a profile. |
| user.bpm.admin.password | Password for the user name above. The default value is admin. This property is needed only if you are creating a profile. |
| user.db2.port | Port for the DB2 database. The default value is 50000. |
| user.db2.instance.username | Linux and UNIX only. DB2 instance user name. The default value is bpminst. |
| user.db2.instance.password | Linux and UNIX only. Password for the user name above. The default value is bpminst1. |
| user.db2.fenced.username | Linux and UNIX only. Fenced user name. The default value is bpmfenc. |
| user.db2.fenced.password | Linux and UNIX only. Password for the user name above. The default value is bpmfenc1. |
| user.db2.das.username | Linux and UNIX only. Administration server (DAS) user name. The default value is bpmadmin. |

Table 44. Keys (continued)

| Key | Description |
|-----------------------|--|
| user.db2.das.password | Linux and UNIX only. Password for the user name above. The default value is bpmadmin1. |

- *logName* is the name of the log file to record messages and results.

Running this command installs the product with the default features. If you want to install specific features or make other changes, see the reference link for the command-line arguments for imcl.

Results

Installation Manager installs the products that are listed and writes a log file to the directory that you specified.

Example

Related reference:

 Command-line arguments for imcl

 WebSphere Enterprise Service Bus system requirements

Installing WebSphere Enterprise Service Bus silently using a response file

You can install WebSphere Enterprise Service Bus by creating a response file and then running a command to use that response file to install the product.

Before you begin

Before you install WebSphere Enterprise Service Bus, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

If you are planning to install WebSphere ESB using DB2 Express with Red Hat Enterprise Linux 6 as a root user, you must ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the `ipcs -l` command.

If you receive the following warning message during the prerequisite checking, use the platform-specific steps below to increase the `ulimit` number.

Current system has detected a lower level of ulimit than the recommended value of recommended_value. Please increase the ulimit number to minimum value of recommended_value and re-start the installation. Shutdown your installer. If you are a root user open a command prompt and issue ulimit -n recommended_value and then restart the installer. If you are a non-root user, work with your system administrator to increase your ulimit -n recommended_value and then restart the installer.

The required value is calculated based on the version of WebSphere Application Server, the feature packs, and the configuration that you are installing.

1. Set the maximum number of open files using the following steps: Linux
 - a. Open `/etc/security/limits.conf`.

- b. Locate the `nofile` parameter and increase the value. If a line containing the `nofile` parameter does not exist, add the following lines to the file:


```
* hard nofile  recommended_value
* soft nofile  recommended_value
```
 - c. Save and close the file.
 - d. Log off and log in again.
2. Restart the computer.
3. Restart the installer.

About this task

If you do not have the prerequisite base products necessary for WebSphere Enterprise Service Bus installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment
- Feature Pack for XML
- Feature Pack for Service Component Architecture (SCA)

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and WebSphere Enterprise Service Bus.

Procedure

To silently install WebSphere Enterprise Service Bus, complete the following steps:

1. Read and accept the license terms before installing. Adding `-acceptLicense` to the command line means that you accept all licenses.
2. Create the response file that will install the required base products and WebSphere Enterprise Service Bus. Copy the sample response file from the following directory to create your own response file:


```
extract_directory/responsefiles/BPM/template_response.xml
```
3. Modify the parameters as directed in the text of the response file template to create your response file. You can also create a response file by recording your actions in Installation Manager. When you record a response file, the selections that you make in Installation Manager are stored in an XML file. When you run Installation Manager in silent mode, Installation Manager uses the data in the XML response file to perform the installation.

Important: Verify that the repository locations at the top of the sample response file point to the correct location in your environment.

4. Run the following command:

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

Administrator or Root user: Windows

```
extract_directory\IM\install.exe -acceptLicense input
extract_directory\responsefiles\productID\template_response.xml -log preferred_log_location\silent_install.log
```

UNIX

Linux

```
extract_directory\IM\installc -acceptLicense input
extract_directory\responsefiles\productID\template_response.xml -log preferred_log_location\silent_install.log
```

Nonadministrator/nonroot user: **Windows**

```
extract_directory\IM\userinstc.exe -acceptLicense input
extract_directory\responsefiles\productID\template_response.xml -log preferred_log_location\silent_install.log
```

UNIX

Linux

```
extract_directory\IM\userinstc -acceptLicense input
extract_directory\responsefiles\productID\template_response.xml -log preferred_log_location\silent_install.log
```

Results

Installation Manager installs any required prerequisites and WebSphere Enterprise Service Bus, and writes a log file to the directory you specified.

Related reference:

 [WebSphere Enterprise Service Bus system requirements](#)

Related information:

 [Installing silently with Installation Manager](#)

 [Recording a response file with Installation Manager](#)

Verifying a stand-alone (qesb) installation

After you have installed WebSphere ESB and created a stand-alone development environment (qesb) profile, you can optionally use the First Steps Console to verify that the product was installed correctly for the profile.

Procedure

1. Start the First Steps console.
2. On the First Steps console, click **Installation verification**.
3. Review the results in the First steps output - installation verification window.
For example:

```
The server name is:qesb
The profile home is:C:\Program Files\IBM\WebSphere\AppServer\profiles\qesb
The profile type is:default
The cell name is:qcell
The node name is:qnode
The current encoding is:Cp1252
The server port number is:9080
CWPIV0020I: The Installation Verification Tool cannot connect to WebSphere Application Server; waiting for the server to start.
CWPIV0010I: Connecting to WebSphere Application Server R8ZMEN0.hursley.ibm.com on port: 9080
CWPIV0020I: The Installation Verification Tool cannot connect to WebSphere Application Server; waiting for the server to start.
ivt.start.commandcmd.exe /c "C:\Program Files\IBM\WebSphere\AppServer\profiles\qesb\bin\startServer.bat" server1 -profileName qesb
>ADMU0116I: Tool information is being logged in file C:\Program
Files\IBM\WebSphere\AppServer\profiles\qesb\logs\server1\startServer.log
>ADMU0128I: Starting tool with the qesb profile
>ADMU3100I: Reading configuration for server: server1
>ADMU3200I: Server launched. Waiting for initialization status.
>ADMU3000I: Server server1 open for e-business; process id is 3220
...
CWPIV0050I: Servlet engine verification status: Passed
...
CWPIV0055I: JavaServer Pages files verification status: Passed
...
CWPIV0060I: Enterprise bean verification status: Passed
...
Health Report
=====

Status  Stand-alone server Node
-----
running server1             qnode
...
CWPIV0070I: The Installation Verification Tool verification succeeded.
CWPIV0080I: The installation verification is complete.
```

Related information:

Starting the First steps console

After you install WebSphere Enterprise Service Bus, you can use the First steps console to verify the installation, start the Profile Management Tool, access product documentation, or direct elements such as servers and administrative consoles related to individual profiles.

Configuring databases

Before starting a profile, you must have configured the databases that are to be used with the profile.

Before you begin

You must have planned your database requirements, including a list of all databases and schema names. For more information, see [Planning your database configuration](#)

Configuring a Microsoft SQL Server database

You can create a stand-alone profile for use with Microsoft SQL Server.

Prerequisites

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

Database restrictions

- The databases that are created for the components must be case-sensitive.

Database privileges and security considerations

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

Table 45 shows the database privileges that are required to access the data store.

Table 45.

| Database management system | Minimum privilege required to use the data store tables | Additional privilege required to create the data store tables |
|----------------------------|--|---|
| Microsoft SQL Server | Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements. | The user ID requires the CREATE TABLE statement privilege. |

Related tasks:



Configuring an existing database during a typical installation

Use the information in this topic to determine the correct database values for configuring your existing database during a typical installation.

“Configuring XA transactions”

You must configure XA transactions after the database is installed and before you start the server. The Microsoft SQL Server JDBC Driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the `SQLServerXADataSource` class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Configuring XA transactions

You must configure XA transactions after the database is installed and before you start the server. The Microsoft SQL Server JDBC Driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the `SQLServerXADataSource` class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

About this task

Failure to configure the XA transactions can result in the following error during server start up: `javax.transaction.xa.XAException:`

```
com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".
```

Procedure

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** dialog box, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
- c. Right-click **Local DTC** and then select **Properties**.
- d. Click the **Security** tab on the **Local DTC Properties** dialog box.
- e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.

- f. Click OK again to close the Properties dialog box, and then close Component Services.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
 - a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from Microsoft Site using URL from Resources section.
 - b. Unzip archive to any folder.
 - c. Copy the sqljdbc_xa.dll file from the JDBC unarchived directory to the Binn directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc_xa.dll file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc_xa.dll file in the x64 folder.
 - d. Execute the xa_install.sql database script on SQL Server . This script installs the extended stored procedures that are called by sqljdbc_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in master database (e.g. for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

Related concepts:

"Configuring a Microsoft SQL Server database" on page 91

You can create a stand-alone profile for use with Microsoft SQL Server.

Creating network deployment environments for use with Microsoft SQL Server

This topic describes how to create a network deployment environment for use with Microsoft SQL Server.

Before you begin

Before creating a profile, complete the following prerequisites:

- Install Microsoft SQL Server on the server that hosts the database.
- If you are going to use the Common Event Infrastructure, you must create the CEI database manually. See *Configuring a Common Event Infrastructure (CEI) database*.

WebSphere ESB packages JDBC drivers for SQL Server. For information about the JDBC drivers (including version and level information), see the page.

Note: You are responsible for providing JDBC driver levels outside of what is packaged with WebSphere ESB.

About this task

You can configure the CommonDB when you create the Deployment Manager profile (WebSphere ESB Advanced only); however, the remaining components must be configured using the Deployment Environment panels in the administrative console. The components to be configured are:

- Common Event Infrastructure
- Business Space

- Messaging Engines

Procedure

1. Create the Deployment Manager profile For more information, see Creating the deployment manager profile.
2. Start the deployment manager using one of the following methods:
 - **Windows** From the **Start** menu, select **IBM > Enterprise Service Bus 7.5 > Profiles > *profile_name* > Start the deployment manager.**
 - In the First Steps console, click **Start the deployment manager.**
 - Use the **startManager** command.
3. Create at least one node (managed profile) for use in the deployment environment. For more information, see “Creating deployment manager and custom profiles using manageprofiles” on page 152.
4. Create the deployment environment:
 - a. In the administrative console, select **Servers > Deployment Environment.**
 - b. Click **New.**
 - c. Provide the information for each step until the step to configure the database.
 - d. On the Database page, update the default values for the components that your environment is using.

Make sure you enter the correct values for the user name and schema name for components below. Deployment Environment configuration does not create any schemas and users as part of configuration. These should exist before the generation of the deployment environment is done. In SQL Server you need to make sure that the default schema for the user is set in the database. It is recommended that for each user you set the same value for the schema in the database - if you do not set the default schema for each user then it would be defaulted to 'dbo' and all the components would get configured with that schema resulting in a non-working environment. The database panel should have values of the schema which correspond to that user. If there is no option to enter a schema value in the field its expected that the default schema which is the same as the user would be set in the database.

- e. Because the component requires manual steps to create the required tables, the Business Space Create Tables check boxes are disabled. Create the tables for this components by following step 6 on page 95.
 - f. Complete the rest of the steps to create the environment and save the settings. You can see **Servers > Deployment Environment** but the deployment environment is not started. Do not start the deployment environment at this time.
5. Optional: If you cleared **Create Tables** when you created the profile in 4, generate the scripts for the message engine.
 - a. In the administrative console, select **Servers > Deployment Environment > *your_deployment_environment* > Deferred Configuration.**
 - b. On the command line, go to where you want to generate the scripts.
 - c. Run **sibDDLGenerator.bat** utility to generate the scripts for each of the schemas required in your environment. For more information about running the utility, see the Deferred Configuration page. The schema names are the values you have chosen in the database panel above.

```
sibDDLGenerator.bat -system sqlserver -version 2005 -platform windows
-schema WPRCM00 -user user_name -statementend ; > output_script_filename
```


Remember to use the correct schema, which is listed in the Deferred Configuration page, and user name. Also, redirect the result to a file. Otherwise, the generated script is printed at the command prompt instead of in a file.

Note: If you configured the databases using a database design file, it is not necessary to run the `sibDDLGenerator.bat` utility. For more information, see “Creating database design files by using the database design tool” on page 97.

6. Manually create the Business Space database:
 - a. In the administrative console, select **Servers > Deployment Environment > *your_deployment_environment* > Deferred Configuration**.
 - b. Find the Business Space scripts.
 - c. Run the `createDatabase_BusinessSpace.sql` script and then the `createTable_BusinessSpace.sql` script.
7. In the administrative console, select **Servers > Deployment Environment > *your_deployment_environment* > Deferred Configuration** and click **Configuration Done**.
8. Log off of the administrative console, shut down the deployment manager, and then shut down all of the custom profiles.
9. Optional: Clean all applicable profile logs or save them in another directory. You may want to clean or move the logs as they will be appended with the last configuration. This can make it difficult to view the most current information.
10. Start the custom profiles, start the deployment manager, and then log in to the administrative console.
11. Start the deployment environment:
 - a. In the administrative console, start the deployment environment by clicking **Servers > Server Types > Deployment Environments**. Select the check box next to the deployment environment and clicking **Start**.
 - b. After 5 to 10 minutes (or longer, depending on the system), refresh the deployment environment page; the Status of the deployment environment changes to **started**.
12. Optional: Check the status of the following items:
 - a. In the administrative console, select **Applications > Enterprise Applications** and check that the installed applications started successfully.
 - b. Select **Resources > JDBC > Data sources** and test that the connection of every component that is not related to the message engine (that is, every component that does not include ME in the name) is successful.

Creating the Common database and tables after profile creation or augmentation

If you postponed creating the Common database and its tables by clearing the **Run database scripts to create database tables (do not select if using a remote database)** check box on the Database configuration panel in the Profile Management Tool, you or your database administrator must create the database and its tables manually. You can do this using scripts that the Profile Management Tool generates during profile creation or augmentation.

Before you begin

This topic assumes that you have performed the following actions:

- You created or augmented a stand-alone server or deployment manager profile using the Profile Management Tool
- In the Database configuration panel in the Profile Management Tool, you entered a name for the database in the **Common database name** or accepted the default common database name CMNDB
- In the Database configuration panel in the Profile Management Tool, you chose to delay creation of the Common database and its tables by clearing the **Run database scripts to create database tables** check box

About this task

Because an installation of WebSphere ESB requires the Common database to function, if you did not allow the Profile Management Tool to create it automatically, you or your database administrator must now create the database and its tables manually by using scripts that the Profile Management Tool generated during the profile creation or augmentation.

Procedure

1. Go to the directory containing the **configCommonDB.sh** script on Linux and UNIX platforms or the **configCommonDB.bat** script on Windows platforms. The default directory to which database scripts are output is:

- **Linux** **UNIX** `profile_root/dbscripts/CommonDB/db_type/db_name`
- **Windows** `profile_root\dbscripts\CommonDB\db_type\db_name`

Note: The Profile Management Tool provides an option to override the default directory. If you selected the option to override the default directory, the location to which database scripts are output is the path you entered in the **Database script output directory** field on the Database Configuration – Part 1 panel.

The variable `db_type` represents the supported database product and `db_name` represents the name of the database.

You must pass the **createDB** parameter to the configCommonDB script if you want to create a new local database; otherwise an existing database will be used.

Note: For Oracle, the batch file creates tables on an existing schema, so the **createDB** parameter should not be specified.

For example:

`configCommonDB.sh createDB` - creates the database and also the tables

`configCommonDB.sh` - creates only the tables and assumes that the database already exists

2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

What to do next

After database creation completes successfully, before starting the server or deployment manager, be sure the database is running even if it is installed locally. Then start the server or deployment manager from the profile's First steps console to ensure there are no errors. You can check the `SystemOut.log` and `SystemErr.log` files for errors. These files are found in the following locations:

- `profile_root/logs/server_name`, for a stand-alone profile

- `profile_root/logs/dmgr`, for a deployment manager profile

Creating database design files by using the database design tool

Use the database design tool to create and generate a design of your database configuration. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of WebSphere ESB.

Creating a database design file for a stand-alone profile or deployment environment by using the database design tool

You can use the database design tool to generate a design file for database tables that can be used by profile creation or when using the deployment environment wizard. The database design tool generates the design file from user-interactive input or from an existing design file.

Before you begin

Ensure that you have installed WebSphere ESB. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator (DBA) or solution architect. Alternatively, it might be a description of required parameters and properties.
- Information about how WebSphere ESB and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

Before you run the database design tool, ensure that you have made the following decisions:

- The type of deployment environment in which the database will be used (stand-alone profile or network deployment environment) based on scalability and high-availability requirements.
- The location of database tables.
- Details about the database type, specifically, but not limited to, the following items:
 - Type of database (DB2, Oracle, DB2 for zOS, SQL Server)
 - Location of the JDBC driver on the system where the server profile will be created
 - User ID and password for authenticating to the database

Tip: Plan for database use when you review information about your planned usage of WebSphere ESB so that you make the necessary decisions on information needed by the database design tool.

About this task

This task describes how to use the database design tool to create a database design file for a stand-alone profile or deployment environment. The input for the database design tool is either user-interactive input or an existing design file. The available options change depending on your environment.

The **DbDesignGenerator** command has the following options.

```
-? , -help
display help info.

-e db_design_file_name
edit the specified database design file (e.g. *.dbdesign, *.properties).

-r db_design_file [-d scripts_output_directory]
when a db_design_file is given, validation will be done on the specified
database design file based on the database scripts.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file] [-d output_directory] ....
[db_design_file] [-d output_directory]
generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Procedure

1. Access the **DbDesignGenerator** command and run the file.

You can find the **DbDesignGenerator** command in the following location:

- **Windows** `install_root\util\dbUtils`

For example, `C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils>DbDesignGenerator.bat`

- **Linux** **UNIX** `/install_root/util/dbUtils`

For example, `/opt/IBM/WebSphere/AppServer/util/dbUtils>DbDesignGenerator.sh`

Tip: If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path. When the database design tool launches successfully, you see the following information:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option (1)Create a database design for Standalone profile or Deployment Environment, type the number 1 and press Enter.

You are prompted to choose a database pattern; for example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology
(2)bpm.advanced.standalone
(3)wesb.nd.topology
(4)wesb.standalone
```

3. To create a database design pattern for the stand-alone profile or deployment environment that you plan to configure, type the number for the appropriate

option and press Enter. For a stand-alone profile, select options that include ".standalone;" for a deployment environment, select options that include ".nd."

For example, to configure the database pattern for a deployment environment for WebSphere Enterprise Service Bus, type the number 3 to select option (3)wesb.nd.topology, and press Enter. You see information similar to the following example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

(1)[CommonDB] WBI_CommonDB : [master] [status = not complete]
(2)[BSPACE] WBI_BSPACE : [status = not complete]
(3)[SibME] WBI_CEI_ME : [status = not complete]
(4)[SibME] WBI_SCA_APP_ME : [status = not complete]
(5)[SibME] WBI_SCA_SYS_ME : [status = not complete]
(6)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. You see the database components that can be configured for the previously selected environment. The database component listed as the master component lists [master] beside the name and must be configured first.

For example, to configure the master component for the (3)wesb.nd.topology design pattern, type the number 1 to select option (1)[CommonDB] WBI_CommonDB : [master] [status = not complete], and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] CommonDB.WBI_CommonDB : : DbType key is not set.
```

Edit this database component? (y/n) [default=y] :

5. To edit the database component and select the database type that you are configuring, type y and press Enter.

After you choose to edit the database component, you see information similar to the following example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

6. Type the number that corresponds to the database type that you want to use for your environment, and press Enter. You obtain a set of prompts to specify the database properties. These prompts vary, depending on the database type that you plan to use.

For example, type the number 1 to select (1)DB2-distributed as the database type. After you select this database type for configuration of the Common DB, you see information similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

(1>false
(2>true

Please enter the number for the Is this database for a Process Center? [default=false] :1
The user ID you use for administrative security[default=] :
The password for the name specified with the adminUserName parameter[default=] :1
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSpace] :
```

7. At each prompt, if a default value is listed, enter the appropriate value for your database configuration, or press Enter to accept the default value.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter 's'; or enter anything else to continue :

8. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the DB2-distributed database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA)
(2)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for (1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA) as the database provider, type the number 1 and press Enter. After you select this database provider, you see information similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, enter the appropriate value for your database configuration, or if a default value is listed, press Enter to accept the default value.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):
```

```
-----
[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BSPACE] WBI_BSPACE : [status = complete]
(3)[SibME] WBI_CEI_ME : [status = complete]
(4)[SibME] WBI_SCA_APP_ME : [status = complete]
(5)[SibME] WBI_SCA_SYS_ME : [status = complete]
(6)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as [status = complete] along with the master component. If this cannot be done for any reason, they remain listed as [status = not complete].

9. Optional: Follow the preceding steps to configure the remaining database components that are listed as [status = not complete]. For any database

components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as [status = complete] as the result of configuring the master database component.

10. When all database components for your database pattern have been configured and are listed as [status = complete] in the database design tool, enter the appropriate number to select [save and exit], and press Enter. For example, after you finish configuring the (3)web.nd.topology database pattern, type the number 6 and press Enter. You see information similar to the following example:

```
[status] web.nd.topology is complete with 0 remaining item(s):
```

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

11. Enter the location where you want to save the database design file, and press Enter. After you enter the location at the prompt, you see information similar to the following example:

```
Please enter the output filename [default=web.nd.topology.dbDesign] :
```

12. Enter the file name for the generated database design file, and press Enter. After you enter the file name at the prompt, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

13. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter.

- a. Specify the full path of the output directory that will contain the scripts for that database component, and press Enter.

After you type y and press Enter to indicate that you want to generate database scripts, you see information similar to the following example for each database component:

```
Please enter the output directory for WBI_CommonDB [default=DB2-distributed-CommonDB] :
```

After you type the location for the output directory and press Enter, you see information similar to the following example after each entry:

```
[info] The script(s) have been generated in C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB
```

After you enter the values for each prompt, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

Results

A database design file and, optionally, database scripts are created at the locations that you specified.

What to do next

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Creating a database design file for a specific component by using the database design tool

You can use the database design tool to generate a design file for database tables required by specific components. The database design tool generates the design file from user-interactive input or from an existing design file.

Before you begin

Ensure that you have installed WebSphere ESB. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator (DBA) or solution architect. Alternatively, it might be a description of required parameters and properties.
- Information about how WebSphere ESB and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

Before you run the database design tool, ensure that you have made following decisions:

- The type of deployment environment in which the database will be used (stand-alone profile or network deployment environment) based on scalability and high-availability requirements.
- The location of database tables.
- Details about the database type, specifically, but not limited to, the following items:
 - Type of database (DB2, Oracle, DB2 for zOS, SQL Server)
 - Location of the JDBC driver on the system where the server profile will be created
 - User ID and password for authenticating to the database

Tip: Plan for database use when you review information about your planned usage of WebSphere ESB so that you make the necessary decisions on information needed by the database design tool.

About this task

This task describes how to use the database design tool to create a database design file for a specific component. The input for the database design tool is either user-interactive input or an existing design file. The available options change depending on your environment.

The **DbDesignGenerator** command has the following options.

```
-? , -help
display help info.

-e db_design_file_name
edit the specified database design file (e.g., *.dbdesign, *.properties).

-r db_design_file [-d scripts_output_directory]
when a db_design_file is given, validation will be done on the specified
database design file based on the database schema.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template dbi generator can be validated.

-g db_design_file [-d output_directory] [db_design_file] [-d output_directory] ...
[db_design_file] [-d output_directory]
generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Procedure

1. Access the **DbDesignGenerator** command and run the file.

You can find the **DbDesignGenerator** command in the following location:

- **Windows** `install_root\util\dbUtils`

For example, **C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils>DbDesignGenerator.bat**

- **Linux** **UNIX** `/install_root/util/dbUtils`

For example, **/opt/IBM/WebSphere/AppServer/util/dbUtils>DbDesignGenerator.sh**

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path. When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

2. To select the option (2)Create a database design for a single component, type the number 2 and press Enter.

You are prompted for a component; for example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bspace
(2)cei
(3)commondb
(4)sca
(5)sibme
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the Common database component, type the number 3 to select option (3)commondb, and press Enter. You see information similar to the following example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

4. Type the number that corresponds to the database type that you want to use for your environment, and press Enter. You obtain a set of prompts to specify the database properties. The prompts vary, depending on the database type that you plan to use.

For example, type the number 1 to select (1)DB2-distributed as the database type. After you select this database type for configuration of the database, you see information similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.  
Database name[default=CMNDB] :  
Database User name[default=] :db2admin  
Schema name[default=] :wesb  
Regular pagesize[default=32k] :  
Regular TableSpace[default=WBISPACE] :  
Temporary pagesize[default=32k] :  
Temporary TableSpace[default=WBITEMPSPACE] :
```

5. At each prompt, enter the appropriate value for your database configuration, or if a default value is listed, press Enter to accept the default value.

After you complete the last prompt, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.  
  
To skip data source properties, enter 's'; or enter anything else to continue :
```

6. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you decided to configure the data source for a database after you selected DB2-distributed as your database type, you see information similar to the following example:

```
[info] Please pick one of the following [database provider(s)] :  
  
(1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA)  
(2)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the number for the appropriate option to select the database provider for the data source, and press Enter. For example, to select the option for (1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA) as the database provider, type the number 1 and press Enter. After you select this database provider for the data source, you see information similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.  
Database server host[default=] :  
Database server port[default=50000] :  
Data source user name[default=] :  
Data source password[default=] :  
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :  
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, if a default value is listed, press Enter to accept the default, or enter the appropriate value for your configuration.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

7. Enter the location where you want to save the database design file, and press Enter. After you enter the location, you see information similar to the following example:
Please enter the output filename [default=CommonDB_DB2-distributed.properties] :
8. Enter the file name for the generated database design file and press Enter. After you enter the file name, you see information similar to the following example:
generate database scripts? (y/n) [default=y] :
9. Optional: If you also want to generate database scripts based on the information provided to the database design tool, perform the following steps:
 - a. Type y and press Enter.
After you type y and press Enter to indicate that you want to generate database scripts, you see information similar to the following example:
Please enter the output directory for CommonDB [default=DB2-distributed-CommonDB] :
 - b. Specify the full path of the output directory that will contain the scripts for that database component, and press Enter.
After you enter the location for the output directory, you see information similar to the following example:
[info] The script(s) have been generated in C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-BPM_ProcessServer
After you enter the values for each prompt, you see information similar to the following example:
[info] thanks, quitting now ...

Results

A database design file is created and, optionally, database scripts are created at the location that you specified.

What to do next

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Troubleshooting the database design tool

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required userName and password properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-  
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```


Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating and configuring the DB2 for z/OS database

If your deployment environment relies on a remote DB2 for z/OS database, use the procedures and reference information in this section to help you configure the database and create the database tables.

Related tasks:

 [Configuring an existing database during a typical installation](#)
Use the information in this topic to determine the correct database values for configuring your existing database during a typical installation.

Create the DB2 for z/OS databases and storage groups using SPUFI, DSNTEP2, or DButility.sh

The profile creation process generates Data Definition Language (DDL) scripts that you can use to create the DB2 database objects for the configuration. There are several tools that you can use to run the DDL scripts to create the database objects for your configuration. You can also use tools such as SPUFI or DSNTEP2 to create and populate the database.

Before you begin

Before you create the DB2 databases and storage groups, you must complete the following tasks:

- Create the server configuration. See *Roadmap: Installing and configuring IBM BPM Express* for information about how to create a configuration for a stand-alone server and network deployment environment.
- Make sure that the DDL has been generated for all the components you want to configure the database with. You can generate the DDL by completing the following tasks:
 - Designing the database configuration
For a network-deployment environment, using the database design tool (DDT) is recommended.
For a stand-alone server environment, the database panels of the Profile Management Tool are usually enough to for stand-alone profiles, although you can use the DDT.
The output of the DDT is a design document (xml file) of the database configuration and, optionally, the SQL scripts to create the database tables.
 - Prepare to use the DDL files

- You might need to copy the DDL files from the WebSphere ESB file system into a partitioned dataset (PDS). You can use a tool such as **Dd12Pds.sh** to copy the files.
- There is no restriction on the naming or organization conventions that apply to the database objects.
- The CEI DDL and the SIB DDL files need to be customized before they can be run.

Note: You can use the sample SIB DDL provided for single database configuration.

About this task

You can run the DDL scripts using **DBUtility.sh**, **SPUFI**, or **DSNTEP2**. You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 objects, particularly in a production environment. The tools can produce an audit trail of the DB2 database commands that have been issued.

If you want to create the database objects across multiple databases but still want to work in the USS environment, you can run the DDL scripts using the **DBUtility.sh** script several times specifying different components for each database name.

If you want to work in the USS environment, you can run the DDL scripts using the **DBUtility.sh** script, which is also supplied with WebSphere ESB.

Important: After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed width MVS data sets.

Procedure

1. Create the databases and storage groups.
2. Populate the databases using the generated DDL scripts. The location of the generated DDL scripts depends on how they were generated.

You can find the **DbDesignGenerator** command in the following location:

- **Windows** `install_root\util\dbUtils`

For example, `C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils>DbDesignGenerator.bat`

- **Linux** **UNIX** `/install_root/util/dbUtils`

For example, `/opt/IBM/WebSphere/AppServer/util/dbUtils>DbDesignGenerator.sh`

For DDL generated by other means, the DDL is in the directories under the following locations:

- `WAS_HOME/profiles/default/dbScripts` for a stand-alone configuration.
- `WAS_HOME/profiles/default/dbScripts` for a network deployment configuration.

Where `WAS_HOME` is the top directory of your WebSphere Application Server configuration.

3. If you are running the DDL from a USS environment, assign the appropriate permissions to the copies of the files; for example:

```
chmod 755 createTable_AppScheduler.sql
```

4. Edit the values in the file to suit your needs. The database names, storage groups and schema names are customized by the product configuration process. Check the values in each file to make sure they match the values that you entered in the response file that provided input to the configuration script and are suitable for your database.

Note: The files can be provided in ASCII format. If the tools that you use to view, edit, and run the scripts require the scripts to be in EBCDIC format, use the **iconv** command to convert the file to EBCDIC. For example:

```
iconv -t IBM-1047 -f ISO8859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

If you have converted the file from ASCII format to EBCDIC but need to run the file in ASCII format, use **iconv** to convert the file back to ASCII. For example:

```
iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

5. Optional: If you want to create database objects outside of the USS environment, for example, by using SPUFI or DSNTEP2, you can use the supplied Ddl2Pds.sh script to copy the customized DDL from USS to a partitioned dataset. For example, to copy the DDL for the WebSphere ESB Common component, enter a command similar to the following from the /usr/lpp/zWESB/V7R5M1/zos.config/samples directory:

```
./Ddl2Pds.sh -Source
/WebSphere/V7S05Z1/AppServer/profiles/default/dbscripts/CommonDB/DB2zOS/S3CELLDB -PDS HEALDR.DDL2PDS.TEST -Component
WPS
```

6. Run the customized scripts using the tool of your choice. For example:

SPUFI A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

DSNTEP2

A sample dynamic SQL program provided with the DB2 for z/OS product.

DBUtility.sh

DBUtility.sh is a utility that is supplied with WebSphere ESB for z/OS and installed in the installation file system. For example:

/usr/lpp/zWESB/V7R5M1/bin/DBUtility.sh. You can use this utility to create the database and storage groups, as well as to run the SQL to create the database tables later, from USS. **DBUtility.sh** uses ASCII input. Here is an example of the syntax used with the **DBUtility.sh** script:

```
/WebSphere/V7S03Z1/AppServer/profiles/default/bin/DBUtility.sh
createTable
-DdbStorageGroup=S3DBST0
-DdbSchemaName=S3CELL
-DsqlScriptName.default=createTable_AppScheduler.sql
-DsqlScriptPath.default=/WebSphere/V7S03Z1/AppServer/profiles/default/dbscripts/CommonDB/DB2zOS/S3CELLDB
/createTable_AppScheduler.sql
-DdbType=DB2UDBS390
-DdbName=S3CELLDB
-DprofileName=default
-DprofilePath=/WebSphere/V7S03Z1/AppServer/profiles/default
-DdbJDBCProperties=/wps/dbscripts/db2v9
-DdbConnectionLocation=DSN810PP
-DdbJDBCClasspath=/usr/lpp/db2910/db2910/jcc/classes
-DdbUserId=wsadmin
-DdbPassword=password
-DdbDelayConfig=false
```

```
-DdbCreateNew=false  
-DdbHostName=winmvsp1.hursley.ibm.com  
-DdbServerPort=448  
>/tmp/output.out 2>>/tmp/error.out
```

7. Verify that the database, storage group, and tables have been created successfully with no errors by inspecting the output.
8. If you are creating a stand-alone configuration, verify the WebSphere ESB installation:
 - a. Start the server.
 - b. Open the administrative console by opening a browser window and typing the URL of the server that you want to view. For example:
`http://server_name.domain_name:port_number/admin`
 - c. Log in to the administrative console.
 - d. Verify that you can see WebSphere ESB for z/OS on the Welcome page. You can click it for more information.
 - e. Navigate around the console to check that the server has a status of started. Also check that all the applications are started, and that the messaging engines are started. If anything has failed to start, you can look in the server job logs for "SEVERE" or "WARNING" messages that provide details about the failure.

Results

The DB2 databases and storage groups are created and populated with the necessary database objects, such as tables and indexes.

What to do next

If you are creating a stand-alone configuration, you can now deploy applications to the server.

If you are creating a network deployment configuration, you must create one or more empty nodes to add to the deployment manager cell. See *Configuring the software after a Custom installation to create one Deployment manager and Custom profiles*.

Granting table privileges to the JCA authentication alias user ID

If the schema name you are using is not the same as the JCA authentication alias user ID you must grant a sub-set of DB2 privileges to the JCA authentication alias user ID.

About this task

The DDL for the Service Integration Bus already contains commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other WebSphere ESB components do not supply any GRANT statements.

Use a schema name that is not the same as the JCA authentication alias to prevent the alias user ID having the power to drop tables. (The power to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to DROP tables.

If you want the WebSphere ESB to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the DDL and editing it to construct GRANT commands from the CREATE commands. Create GRANT commands like:

```
GRANT ALL PRIVILEGES ON TABLE  
cell.tablename TO userid/sqlid
```

Where *userid/sqlid* is the JCA authentication alias user ID.

Setting the correct schema name for the SIBs

To ensure the SIB messaging engines can access the appropriate DB2 tables, set the correct schema name for the SIB messaging tables to use to access the DB2 tables.

Before you begin

Start the server (stand-alone server or deployment manager).

About this task

Use the administrative console to change the schema names.

Procedure

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each bus:
 - a. Select **Messaging engines**, then click the name that is displayed.
 - b. Click **Message store**.
 - c. Change the value of **Schema name** to the name used when creating the DB2 tables for this SIB.
 - d. Click **Apply**.
 - e. Save your configuration changes.
4. Log out of the administrative console.
5. Stop, then restart the server.
6. Look in the output of the Adjunct job log for successful SIB messaging engine startup messages. For example:

```
BB000222I: "BusName"  
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

Results

The schema name used by the SIB messaging tables to access the DB2 tables is changed.

Verifying the installation with DB2 for z/OS

When verifying an installation with a DB2 for z/OS database, it is important to check the Servant and Adjunct job logs to see whether there are any error messages that might indicate problems accessing the data store.

Procedure

1. Ask your DB2 system administrator to check the authorities that have been granted to ensure that you have not granted more authority than necessary to

any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration.

2. Ask your DB2 system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over to people to use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.
3. Log in to the administrative console.
4. In the administrative console, check that all the applications are started, the messaging engines are started, and all the data sources can be accessed using the **Test Connection** option. If any application has failed to start, look in the Servant and Adjunct job logs for SEVERE or WARNING messages that provide detail about the failure.
 - If you see DB2 errors such as SQLCODE -204, in the administrative console, set the correct schema name or currentSQLID value in the custom properties section of the data sources. If the schema name is not the same as the user ID in the JCA authentication aliases, the SQL requests try to find tables qualified by the user ID in the JCA authentication alias.
 - If you see DB2 deadlock errors such as SQLCODE -913 Reason Code 00C90088, set the RRULOCK DB2 parameter to YES to prevent tablespace locks on WebSphere ESB tables.

What to do next

If all the messaging engines have initialized correctly, and you do not see any other errors related to opening JDBC connections, you can continue to customize your configuration of WebSphere ESB.

Modifying the transaction log options for a DB2 database

When you configure DB2 for use with WebSphere ESB, you must modify the transaction log options.

Procedure

1. Start a DB2 command line processor.
2. Run the following commands:

```
CONNECT TO [DB_name]
UPDATE DB CFG FOR DB_name USING LOGFILSIZ 4096 IMMEDIATE
UPDATE DB CFG FOR DB_name USING LOGSECOND 64 IMMEDIATE
CONNECT RESET
```
3. Stop and restart DB2.

Configuring WebSphere Enterprise Service Bus

You can configure WebSphere ESB to form a stand-alone or network deployment environment.

The configuring phase consists of two types of tasks: product configuration tasks and environment configuration tasks. Product configuration tasks are for setting up the product profiles and configuring the database, while environment configuration tasks are for setting up and generating the deployment environment.

One of the environment configuration tasks is creating the clusters of the deployment environment. Clusters in a deployment environment require specific tables, schemas, and user permissions based on the functional purpose of the cluster. For example, a deployment environment could include a messaging infrastructure cluster that accesses messaging engine database tables.

Figure 8 illustrates the task flow for planning, installing, and configuring the product and the environment.

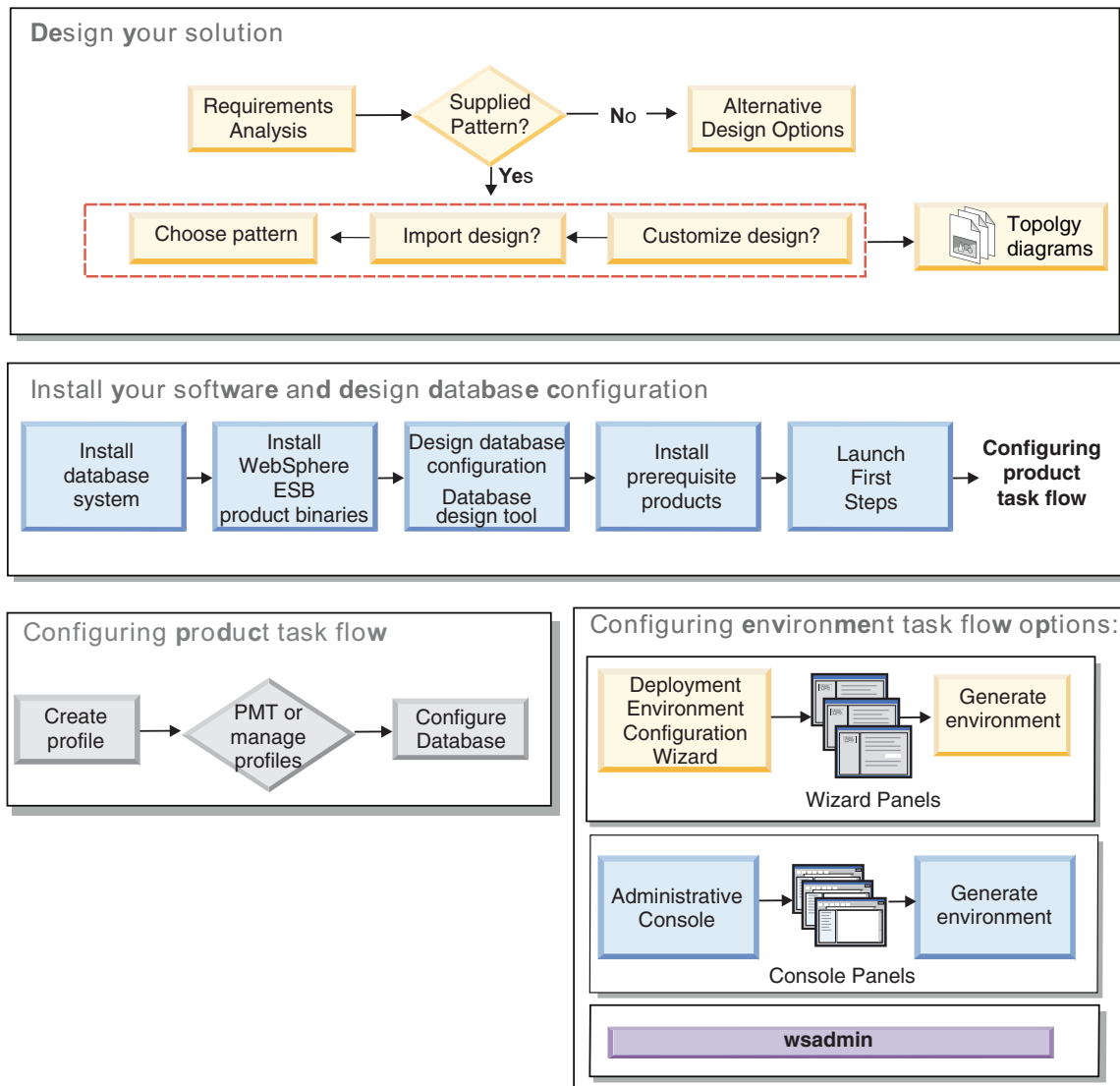


Figure 8. Task flow for planning, installing, and configuring the product and the environment

Creating and augmenting profiles

After you install the product, you must create one or more profiles to define the runtime environment.

Configuration prerequisites and considerations

Before you configure the software for WebSphere ESB, review the prerequisites and other considerations.

Prerequisites for creating or augmenting profiles:

Before creating or augmenting a profile, you must ensure that a series of prerequisites have been met.

- You must have an existing installation of WebSphere ESB. If you do not, see *Installing and configuring WebSphere ESB for installation procedures*.
- If you are not the user ID who installed the product, you must have write permission to selected directories within the WebSphere ESB installation. See “Granting write permission of files and directories to nonroot users for profile creation” for instructions on how to obtain these permissions. You must create your profiles in a directory other than *install_root/profiles*.
- **Windows** To install or run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the `pmt.bat` file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the `install_root\bin\ProfileManagement` directory:

```
runas /user:MyAdminName /env pmt.bat
```

Non-administrative users are prompted for the administrator password.

- **Windows** To install or run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the `install_root\bin` directory:

```
runas /env /user:MyAdminName "manageprofiles -create -templatePath install_root/profileTemplates/BPM/default.procctr"
```

Non-administrative users are prompted for the administrator password.

- You cannot use the Profile Management Tool to create or augment profiles on WebSphere ESB installations on 64-bit architectures except on the Linux on zSeries® platform. To create or augment profiles on other 64-bit architectures, you can use the **manageprofiles** command-line utility.

Note: You can use the Profile Management Tool on these architectures if you use a WebSphere ESB 32-bit installation.

- You must shut down any servers associated with a profile you plan to augment.
- You must review Naming considerations for profiles, nodes, servers, hosts, and cells for information about reserved terms and issues that you must consider when naming your profile, node, host, server (if applicable), or cell (if applicable).
- You must have enough disk and temporary space to create or augment the new profile. For information about space requirements, see the detailed system requirements web page, and then select the link to your version of WebSphere ESB.
 - <http://www.ibm.com/software/integration/wsesb/sysreqs/>

Granting write permission of files and directories to nonroot users for profile creation:

The product installer (who can be a root/Administrator or nonroot user) can grant write permission to the appropriate WebSphere ESB files and directories to nonroot users. The nonroot users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

The following example task shows how to create a group that is authorized to create profiles.

Throughout this text, the terms "installer" and "product installer" refer to the user ID that installed WebSphere ESB.

Restriction: WebSphere ESB does not support changing ownership of existing profiles from the product installer to nonroot users. Thus, profile augmentation by nonroot users of profiles owned by another user is not supported.

Nonroot users create their own profiles so that they can manage their own environments. Typically, they manage environments for development purposes.

Nonroot users must store their profiles in their private directory structure, not in the *install_root*/profiles directory of the product.

Restriction: An ease-of-use limitation exists for nonroot users who create profiles. Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonroot users. The nonroot user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonroot users a range of values for each of the fields, and assign responsibility to the nonroot users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, then certain directories and files were created. Because these directories and files were created, skip the steps in this topic that create these directories and files. If no profile was previously created, then you must complete the steps to create the required directories and files. In most cases, a profile has been created previously.

Steps the product installer must perform to grant appropriate permissions

The installer can perform the following steps to create the profilers group and give the group appropriate permissions to create a profile.

1. Log on to the WebSphere ESB system as the product installer. (The product installer can be a root/Administrator or nonroot user.)
2. Using operating system commands, perform the following steps:
 - Create a group named profilers, which will contain all users who can create profiles.
 - Create a user named user1, who can create profiles.
 - Add users product_installer and user1 to the profilers group.
3. Linux UNIX Log off and log back on as the installer to pick up the new group.
4. Create the following directories as the installer if no profile exists:
 - Linux UNIX Create the *install_root*/logs/manageprofiles directory:
`mkdir install_root/logs/manageprofiles`

Windows Create the *install_root*\logs\manageprofiles directory by following instructions in the Windows documentation. For this example procedure, the directory is:

install_root\logs\manageprofiles

- **Linux** **UNIX** Create the *install_root*/properties/fsdb directory:
`mkdir install_root/properties/fsdb`

Windows Create the *install_root*\properties\fsdb directory by following instructions in the Windows documentation. For this example procedure, the directory is:

install_root\properties\fsdb

5. As the installer, follow directions for your operating system to create the profileRegistry.xml file if no profile exists. For this example, the file paths are:

Linux

UNIX

install_root/properties/profileRegistry.xml

Windows

install_root\properties\profileRegistry.xml

Follow instructions for your operating system to add the following information to the profileRegistry.xml file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions.

Linux

UNIX

The following example assumes that the variable \$WASHOME is the WebSphere ESB root installation directory /opt/IBM/WebSphere/ESB.

```
export WASHOME=/opt/IBM/WebSphere/ESB
echo $WASHOME
echo "Performing chgrp/chmod per WAS directions..."
chgrp profilers $WASHOME/logs/manageprofiles
chmod g+wr $WASHOME/logs/manageprofiles
chgrp profilers $WASHOME/properties
chmod g+wr $WASHOME/properties
chgrp profilers $WASHOME/properties/fsdb
chmod g+wr $WASHOME/properties/fsdb
chgrp profilers $WASHOME/properties/profileRegistry.xml
chmod g+wr $WASHOME/properties/profileRegistry.xml
chgrp -R profilers $WASHOME/profileTemplates
```

HP-UX Issue the following additional command where *profile_template_name* is default, dmgr, or managed:

```
chmod -R g+wr $WASHOME/profileTemplates/profile_template_name/documents
```

HP-UX The ownership of files is preserved when the files are copied to the profile directory during profile creation. You granted write permission to the profile directory so that files copied to the profile directory can be modified as part of the profile creation process. Files that are already in the profileTemplates directory structure before the start of profile creation are not modified during profile creation.

Linux

Issue the following additional commands:

```
chgrp profilers $WASHOME/properties/Profiles.menu
chmod g+wr $WASHOME/properties/Profiles.menu
```

Windows

The following example assumes that the variable \$WASHOME is the WebSphere ESB root installation directory C:\Program Files\IBM\WebSphere\

ESB. Follow instructions in the Windows documentation to give the profilers group read and write permission to the following directories and their files:

```
@WASHOME\logs\manageprofiles
@WASHOME\properties
@WASHOME\properties\fsdb
@WASHOME\properties\profileRegistry.xml
```

You might have to change the permissions on additional files if the nonroot user encounters permission errors. For example, if the product installer authorizes a nonroot user to delete a profile, then the product installer might have to delete the following file:

```
Linux      UNIX      install_root/properties/profileRegistry.xml_LOCK
Windows    install_root\properties\profileRegistry.xml_LOCK
```

Give write access to the nonroot user for the file to authorize the user to delete the file. If the nonroot user still cannot delete the profile, then the product installer can delete the profile.

Result

The installer created the profilers group and gave the group proper permissions to certain directories and files to create profiles. These directories and files are the only ones in the installation root of WebSphere ESB to which a nonroot user needs to write to create profiles.

What to do next

The nonroot user that belongs to the profilers group can create profiles in a directory that the nonroot user owns and to which the nonroot user has write permission. However, the nonroot user cannot create profiles in the installation root directory of the product.

A nonroot user ID can manage multiple profiles. The same nonroot user ID can manage an entire profile, whether it is the deployment manager profile, a profile that contains the servers and the node agent, or a custom profile. A different user ID can be used for each profile in a cell, whether global security or administrative security is enabled or disabled. The user IDs can be a mix of root and nonroot user IDs. For example, the root user might manage the deployment manager profile, while a nonroot user might manage a profile that contains servers and the node agent, or vice versa. However, typically, a root user or a nonroot user can manage all profiles in a cell.

The nonroot user can use the same tasks to manage a profile that the root user uses.

Database prerequisites for creating or augmenting profiles:

Before creating or augmenting a profile, you must ensure that a series of database prerequisites have been met.

The following prerequisites relate to product databases:

- During the profile creation or augmentation process, you configure the Common database used by selected components. Whether you plan to create new databases and tables or postpone actual database configuration by producing scripts that must be run manually by you or your database administrator (DBA), you must know the database details listed in the following topic:

- manageprofiles parameters
- If you plan to use or create the Common database on a remote server, you must have created the database before beginning to create or augment the profile. You can create a database on the local server or use an existing one on a remote server. For information about the default scripts you can use to create the database, see *Creating the Common database manually before product installation*.
- If you plan to use DB2 on a remote z/OS workstation for the Common database repository, your DBA must create, on the z/OS server, the database called CMNDB, and the correct storage group for that database. The DBA can use the site's standard database definition tools and procedures.

Before running **CreateDB.sh**, you must allocate the following buffer pools with these DB2 commands:

```
-ALTER BUFFERPOOL (BP1) VPSIZE(20000)
-ALTER BUFFERPOOL (BP2) VPSIZE(20000)
-ALTER BUFFERPOOL (BP3) VPSIZE(20000)
```

You must also make sure that permission to use them has been granted as follows:

```
GRANT USE OF BUFFERPOOL BP1 TO PUBLIC;
GRANT USE OF BUFFERPOOL BP2 TO PUBLIC;
GRANT USE OF BUFFERPOOL BP3 TO PUBLIC;
```

- Database administrator (DBA) privileges are required for the database configuration panels that are part of creating a deployment manager profile. If the user ID does not have DBA privileges, use this workaround:
 1. Install the product without creating a profile.
 2. Use the Profile Management Tool to create the deployment manager and the custom profiles using the Advanced path for all. Do not use the Typical path. Do not create database tables as part of the profile creation process.
 3. Have the DBA create the Common DB. The information at the following site provides the necessary scripts to manually create database objects: .
 4. Federate the custom profiles to the deployment manager.
 5. Using the administrative console, create the required deployment environment. See *Creating a deployment environment using a pattern* for more information.

- Linux UNIX If you plan to use DB2 Universal Database:

You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands, which are used during profile creation. Add the **db2profile** script to the /etc/profile directory:

vi /etc/profile and add below lines:

```
export PATH=/opt/IBM/db2/V9.7/bin:$PATH
. /home/db2inst1/sqllib/db2profile
```

After adding the db2profile script to the /etc/profile directory, you must run the **db2profile** script to set the DB2 environment.

You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using db2inst1 as the user ID, add the root to the /etc/group administrative groups:

vi /etc/group and update below lines:

```
dasadm:|:101:dasusr1,db2inst1,root
db2iadm:|:102;root
db2fadm:|:103;db2fenc1,root
```

Typical profile creation Exceptions:

When the **db2profile** script is not run:

```
/opt/RL3/wps4013/utl1/dbutils/profilehelpers/commonDButils: ant:DB1: Execute failed:
java.io.IOException: Cannot run program "db2" (in directory "/opt/RL3/
wps4013/profiles/dbmgr01/dbscripts/CommonDB/DB2/WPSDB1")
```

When the DB2 database manager is not running:

SQL1032N No start database manager command was issued. SQLSTATE=57019

When the user who installed WebSphere ESB and is creating the profile is not added to the DB2 administrative groups:

SQL1092N "ROOT" does not have the authority to perform the requested command.

When DB2 database manager is down or not running...

SQL1032N No start database manager command was issued. SQLSTATE=57019

- If you plan to use Microsoft SQL Server 2005 or Microsoft SQL Server 2008 with a standalone profile, and will put the messaging engine tables in the Common Database, then you must perform the following steps:
 1. Manually add four schemas to the Common database before creating stand-alone server profiles. These schemas are XXXSS00, XXXSA00, XXXCM00, and XXXBM00, where XXX is the first three characters of the name of the Common database.
 2. The following command configures the Messaging Engines on SQL Server with the schemas that were defined above. The command uses the dbUserId and dbPassword that you specified for CommonDB.



For Microsoft SQL Server JDBC 1.2 driver

For Microsoft SQL Server JDBC 2.0 driver

Starting the Profile Management Tool:

Before you start the Profile Management Tool, be aware of the restrictions and ensure that certain prerequisites are met. You can start the Profile Management Tool in several ways, depending on the platform on which it is running.

Restrictions:

- You cannot use the Profile Management Tool to create or augment profiles on WebSphere ESB installations on 64-bit architectures except on the Linux on zSeries platform. To create profiles on other 64-bit architectures, you can use the **manageprofiles** command-line utility. For information about using the **manageprofiles** command-line utility, see “Creating profiles using the manageprofiles command-line utility” on page 150. You can also use the Profile Management Tool on WebSphere ESB 32-bit installations on these architectures.
-   **Restriction for nonadministrative users with multiple instances:** If you install multiple instances of WebSphere ESB as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a com.ibm.wsspi.profile.WSProfileException or Access is denied message occurs in the *install_root\bin\ProfileManagement\pmt.bat* file. By default, nonadministrative users do not have access to the Program Files directory, which is the default installation location for the product. To resolve

this issue, nonadministrative users either install the product by themselves or be given permission to access the other product instances.

Linux **UNIX** **Windows** The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, then English is used. You can override the default language by starting the Profile Management Tool from the command line and using the `java user.language` setting to replace the default language. Run the following command:

- **Linux** **UNIX** `install_root/java/bin/java -Duser.language=locale install_root/bin/ProfileManagement/startup.jar`
- **Windows** `install_root\java\bin\java -Duser.language=locale install_root\bin\ProfileManagement\startup.jar`

For example, to start the Profile Management Tool in the German language on a Linux system, type the following command:

```
install_root/java/bin/java -Duser.language=de install_root/ \
bin/ProfileManagement/startup.jar
```

Starting the tool on all platforms

Start the tool on any platform from the First steps console. See for more information.

Starting the tool on Linux and UNIX platforms

Linux **UNIX** You can start the tool on Linux and UNIX platforms by running the command `install_root/bin/ProfileManagement/pmt.sh`

Linux On Linux platforms only, you can also use operating system menus to start the Profile Management Tool. For example, click `Linux_operating_system_menus_to_access_programs > IBM WebSphere > your_product > Profile Management Tool`.

Starting the tool on Windows platforms

Windows You can use the following methods to start the tool on Windows platforms:

- Use the Windows Start menu. For example, click **Start > Programs or All Programs > IBM WebSphere > Enterprise Service Bus 7.0 > Profile Management Tool**.
- Run the command `install_root\bin\ProfileManagement\pmt.bat`

Creating stand-alone profiles using the Profile Management Tool

You can use the Profile Management Tool to create the profile for a stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

Before you begin

- Review Prerequisites for creating or augmenting profiles.
- **Solaris** When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile

Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0  
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

Tip: Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create profiles, as described in *Configuring the software using command-line utilities and wsadmin*.

Procedure

1. Start the Profile Management Tool. For methods to start this tool, see *Starting the Profile Management Tool*. The Welcome page displays.
2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
The **Profiles** tab displays.
The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.
3. In the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.
4. On the Environment Selection page, select the profile to be created.
 - a. Select **WebSphere Enterprise Service Bus > Stand-alone enterprise service bus**
 - b. Click **Next**The Profile Creation Options page displays.
5. On the Profile Creation Options page, create the stand-alone profile using one of the following options:

Typical profile creation

Creates a profile with default configuration settings.

Restriction: If you plan to federate the stand-alone server profile to a deployment manager, do not use the **Typical profile creation** option to create the profile. The default values for messaging engine storage and database type provided in a **Typical** profile creation are not suitable deployment environment installations. Instead, use the **Advanced profile creation** option to create the profile.

Advanced profile creation

Creates a profile using the configuration values you specify.

Table 46 on page 121 provides more information about the options for creating a stand-alone profile.

Table 46. Selecting the creation option for your stand-alone profile

| Select | When you want to . . . |
|----------------------------------|--|
| Typical profile creation | <p>Allow the Profile Management Tool to perform the following actions:</p> <ul style="list-style-type: none"> • Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell. • Install the administrative console. • Create a personal security certificate for the profile. The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year. • Create a root signing security certificate for signing other certificates. The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years. • Create a system service to run the server. Applicable only when your operating system and the privileges of your user account permit. • Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database. • Select any of the supported database products and set the database configuration for the CommonDB. |
| Advanced profile creation | <ul style="list-style-type: none"> • Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable). • Deploy the administrative console. • Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets). • Create a web server definition. • Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services. |

What to do next

Continue creating one of the following types of stand-alone profiles:

- **Typical profile creation**
- **Advanced profile creation**

Creating a typical stand-alone profile with the Profile Management tool:

You can use the Profile Management Tool to create the profile for a typical stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

Before you begin

Complete the initial configuration steps provided in “Creating stand-alone profiles using the Profile Management Tool” on page 119.

Procedure

1. Optional: Optionally enable administrative security. You can enable administrative security now, or later from the administrative console.

- To enable administrative security now, leave the **Enable administrative security** check box selected, and supply a user name and password to be used later to log on to the administrative console.
- To disable administrative security, clear the check box.

Click **Next** to continue.

2. On the Database Configuration page, perform the following actions:
 - From the **Select a database product** drop-down, select the database product to be used by the profile.
 - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

If the database product you are using with the software already exists, select **Use an existing local or remote database**.

- Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

If you do not select the checkbox, the scripts are output to the default directory.

- Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

3. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. See the Database configuration fields for Profile Management Tool configuration information for details about each required field.

You can configure parameters for the following databases:

DB2 After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

DB2 Universal Database for z/OS

You cannot create a new database using DB2 Universal Database for z/OS. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

Microsoft SQL Server

After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Database Configuration - Part 3 page. When you have completed the configuration, click **Next** to go to the Profile Summary page.

Oracle You cannot create a new database using this database. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Database Configuration - Part 3 page. When you have completed configuration on the Database Configuration - Part 3 page, then click **Next** to go to the Profile Summary page.

DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox)

After setting the values for the DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox) database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

4. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message The Profile Management tool created the profile successfully.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- The Profile Management tool created the profile but errors occurred, which indicates that profile creation completed but errors were generated.
- The Profile Management tool cannot create the profile, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.

You have completed the steps to create the profile with default configuration settings.

What to do next

From the First Steps Console, you can start the stand-alone profile you have just created.

If you did not run the database scripts during profile creation, you must load the database with system information after the database has been created.

Creating an advanced stand-alone profile with the Profile Management tool:

You can use the Profile Management Tool to create an advanced profile for a stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

Before you begin

Complete the initial configuration steps provided in “Creating stand-alone profiles using the Profile Management Tool” on page 119.

Procedure

1. In the Optional Application Deployment page, select the applications you want to deploy to the profile environment.

Deploy the Sample applications

Installs the WebSphere ESB and WebSphere Application Server sample applications. The sample applications are not recommended for deployment to production environments.

Deploy the administrative console (recommended)

Installs a web-based administrative console that manages the server.

Deploy the default application

Installs the default application that contains the Snoop, Hello, and HitCount Servlets.

2. In the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value.

Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Naming considerations for profiles, nodes, servers, hosts, and cells for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse...** button to go to the profile directory.

The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform. The following examples show the platform differences and *profile_name* is the name you specify:

- Linux `install_root/profiles/profile_name`
- Windows `install_root\profiles\profile_name`

An error message is displayed if any of the following issues occur:

- The *profile_name* you specify is not unique.
 - The directory you specify is not empty.
 - Your user ID does not have sufficient permissions for the directory.
 - There is not sufficient space to create the profile.
- c. Optional: Select the **Make this profile the default** check box to make the profile you are creating the default profile.

Note: This check box is shown only if you have an existing profile on your system.

When a profile is made to be the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See Profile commands in a multiprofile environment for more information.

- d. From the **Server runtime performance tuning setting** pull-down list, select a performance tuning level appropriate for the profile you are creating.
- e. Click **Next**.

Note: If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

3. In the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value.

Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next** to display the Administrative Security page.

4. Optional: Optionally enable administrative security.

You can enable administrative security now, or later from the administrative console. To enable administrative security now, leave the **Enable administrative security** check box selected, supply a user name and password to log on to the administrative console, and click **Next**. To disable administrative security, clear the check box.

5. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

Perform the following actions:

- To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.
- To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

6. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.

- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower[®] signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

An imported certificate is added to the `key.p12` file or the `root-key.p12` file.

If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

7. From the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**.

The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

- Linux UNIX `profile_root/properties/portdef.props`
- Windows `profile_root\properties\portdef.props`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see the topic Updating ports in an existing profile in the WebSphere Application Server Network Deployment information center. Run the `updatePorts.ant` file through the **ws_ant** script detailed in this topic.

| If you are installing | Next step |
|--|---|
| On a Linux or Windows platform, <i>and have root or Administrator group privileges</i> | The Linux or Windows Service Definition page is displayed. Proceed to step 8 on page 127. |
| On any other platform or as a nonroot user on a Linux or Windows platform | The Web Server Definition page is displayed. Proceed to step 9 on page 128. |

8. In the Service Definition page, indicate whether or not to use a Windows service or Linux service to run WebSphere ESB

Windows The Windows Service Definition page displays for the Windows platform only when the ID that installs the Windows service has the Administrator group privilege. If the profile is configured as a Windows service, the product starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Windows service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

Important: If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, it must belong to the Administrator group, and it must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

IPv6 considerations when running profiles as Windows services

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the WebSphere ESB Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the WebSphere ESB Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

Linux The Linux Service Definition page is displayed only if the current operating system is a supported version of Linux and the current user has the appropriate permissions.

WebSphere ESB attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Linux service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

By default, WebSphere ESB is not selected to run as a Linux service.

To create the service, the user who runs the Profile Management Tool must be the root user. If you run the Profile Management Tool with a non-root user ID, the Linux Service Definition page is not displayed, and no service is created.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have proper privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

Click **Next** to continue

9. To include a Web server definition in the profile now, perform the following steps:
 - a. Select the **Create a Web server definition** check box.
 - b. Specify the Web server characteristics on the page, and click **Next**
 - c. Specify the Web server characteristics on Part 2 of the page.

If you use a Web server to route requests to WebSphere ESB, you need to include a Web server definition. You can include the definition now, or define the Web server to WebSphere ESB later. If you define the Web server definition during the creation of this profile, you can install the Web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the Web server to WebSphere ESB after you create this profile, you must define the Web server in a separate profile.
 - d. Click **Next**.
10. On the Business Space Configuration page, leave the **Configure Business Space** check box selected to set up Business Space, an integrated user experience for application users across the IBM business process management portfolio. If you want to configure IBM Forms Server to work with Human Task Management widgets in Business Space, select the **Configure IBM Forms Server** check box and enter the HTTP location of the IBM Forms Server translator and IBM Forms Server installation root. Then click **Next**.
11. Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.
 - e. Click **Next**.
12. If you chose not to configure the databases using a design file, specify the database details on the database configuration panels. For details, see the following steps:
 - a. On the Database Configuration page, perform the following actions:
 - From the **Select a database product** drop-down, select the database product to be used by the profile.
 - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

If the database product you are using with the software already exists, select **Use an existing local or remote database**.
 - Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

If you do not select the checkbox, the scripts are output to the default directory.

- Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.
- b. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. See the Database configuration fields for Profile Management Tool configuration information for details about each required field.

You can configure parameters for the following databases:

DB2 After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

DB2 Universal Database for z/OS

You cannot create a new database using DB2 Universal Database for z/OS. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

Microsoft SQL Server

After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Database Configuration - Part 3 page. When you have completed the configuration, click **Next** to go to the Profile Summary page.

Oracle You cannot create a new database using this database. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Database Configuration - Part 3 page. When you have completed configuration on the Database Configuration - Part 3 page, then click **Next** to go to the Profile Summary page.

DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox)

After setting the values for the DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox) database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

13. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

14. Manually configure the SMTP server to enable mail notifications. Refer to Configuring the SMTP server.

What to do next

From the First Steps Console, you can start the stand-alone profile you have just created.

If you did not run the database scripts during profile creation, you must load the database with system information after the database has been created.

Creating the deployment manager profile

You can use the Profile Management Tool to create the deployment manager profile of your network deployment configuration.

Before you begin

- Prerequisites for creating or augmenting profiles.
- **Solaris** When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
After adding the lines, run the following command before starting the Profile Management Tool:
`xrdb -load user_home/.Xdefaults`

Tip: Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create profiles, as described in Configuring the software using command-line utilities and `wsadmin`.

About this task

When creating a network deployment environment, the deployment manager profile is the first profile that you create. The deployment manager administers one or more nodes, for which you create custom profiles.

Procedure

1. Start the Profile Management Tool. For methods to start this tool, see Starting the Profile Management Tool. The Welcome page displays.
2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.
The **Profiles** tab displays.
The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.
3. In the **Profiles** tab, click **Create**.
The Environment Selection page opens in a separate window.

4. On the Environment Selection page, select the profile to be created.
 - a. Select **WebSphere Enterprise Service Bus > Enterprise service bus deployment manager**
 - b. Click **Next**

The Profile Creation Options page displays.

5. From the Profile Creation Options page, decide whether to create the deployment manager profile using the **Advanced** or **Typical** option.
 - The **Typical profile creation** option creates a profile with default configuration settings.
 - The **Advanced profile creation** option lets you specify your own configuration values for a profile.

Results

You have created the deployment manager profile for the network deployment configuration.

What to do next

Create and configure one or more custom profiles (managed nodes) for your network deployment configuration.

Creating a typical deployment manager profile:

You can use the Profile Management Tool method to create a deployment manager profile of your network deployment configuration.

Before you begin

Complete the initial steps in Creating the deployment manager profile.

About this task

Create a typical stand-alone profile to allow the Profile Management Tool to perform the following functions:

- Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
- Install the administrative console.
- Create a personal security certificate for the profile.

The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year.

- Create a root signing security certificate for signing other certificates.

The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years.

- Create a system service to run the server.

Applicable only when your operating system and the privileges of your user account permit.

- Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.
- Select any of the supported database products and set the database configuration for the CommonDB.

Procedure

1. Optionally enable administrative security. You can enable administrative security now, or later from the administrative console.
 - To enable administrative security now, leave the **Enable administrative security** check box selected, supply a user name and password to be used later to log on to the administrative console.
 - To disable administrative security, clear the check box.

Click **Next** to continue.

2. On the Database Configuration page, perform the following actions:
 - From the **Select a database product** drop-down, select the database product to be used by the profile.
 - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

If the database product you are using with the software already exists, select **Use an existing local or remote database**.

- Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

If you do not select the checkbox, the scripts are output to the default directory.
 - Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.
3. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. For information about each required field, see Database configuration fields for Profile Management Tool configuration.

You can configure parameters for the following databases:

DB2 After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

DB2 Universal Database for z/OS

You cannot create a new database using DB2 Universal Database for z/OS. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

Microsoft SQL Server

After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Database Configuration - Part 3 page. When you have completed configuration on the Database Configuration - Part 3 page, click **Next** to go to the Profile Summary page.

Oracle You cannot create a new database using this database. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Database Configuration - Part 3 page. When you have completed configuration on the Database Configuration - Part 3 page, click **Next** to go to the Profile Summary page.

DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox)

After setting the values for the DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox) database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

4. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the following message: **The Profile Management tool created the profile successfully.**

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

The Profile Management tool augmented the profile but errors occurred

Indicates that profile augmentation completed but errors were generated.

The Profile Management tool cannot augment the profile

Indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

5. Proceed to the First Steps Console.
 - a. Ensure that the **Launch First Steps Console** checkbox is checked.
 - a. Click **Finish**.
 - b. Close the Profiles page, which is open in a separate window.
- Use the First steps console to access the product documentation. You have completed the steps to create the profile with default configuration settings.

Results

You have created the deployment manager profile for the network deployment configuration.

What to do next

Create and configure one or more custom profiles (managed nodes) for your network deployment configuration.

Creating an advanced deployment manager profile:

You can use the advanced Profile Management Tool method to create a deployment manager profile of your network deployment configuration.

Before you begin

Complete the initial steps in Creating the deployment manager profile.

About this task

Use the advanced method to specify the configuration values for a profile. Use the advanced method to create a stand-alone profile when you want to:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Deploy the administrative console.
- Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

Procedure

1. In the Optional Application Deployment page, select whether to deploy the administrative console.

Click **Next** to display the Profile Name and Location page.

2. In the Profile Name and Location page, perform the following steps:

- a. In the **Profile name** field, specify a unique name or accept the default value.

Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Naming considerations for profiles, nodes, servers, hosts, and cells for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse...** button to go to the profile directory.

The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform. The following examples show the platform differences and *profile_name* is the name you specify:

- **Linux** **UNIX** `install_root/profiles/profile_name`
- **Windows** `install_root\profiles\profile_name`

An error message is displayed if any of the following issues occur:

- The *profile_name* you specify is not unique.
 - The directory you specify is not empty.
 - Your user ID does not have sufficient permissions for the directory.
 - There is not sufficient space to create the profile.
- c. Optional: Select the **Make this profile the default** check box to make the profile you are creating the default profile.

Note: This check box is shown only if you have an existing profile on your system.

When a profile is made to be the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the bin directory in the product installation root.

When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See Profile commands in a multiprofile environment for more information.

- d. From the **Server runtime performance tuning setting** pull-down list, select a performance tuning level appropriate for the profile you are creating.
- e. Click **Next**.

Note: If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

3. In the Node, Host and Cell Names page, perform the following actions for the profile you are creating:
 - In the **Node name** field, enter a name for the node or accept the default value.

Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.
 - In the **Server name** field, enter a name for the server or accept the default value.
 - In the **Host name** field, enter a name for the host or accept the default value.
 - In the **Cell name** field, enter a name for the cell or accept the default value.
4. Optionally enable administrative security. You can enable administrative security now, or later from the administrative console.
 - To enable administrative security now, leave the **Enable administrative security** check box selected, supply a user name and password to be used later to log on to the administrative console.
 - To disable administrative security, clear the check box.

Click **Next** to continue.

5. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

Perform the following actions:

- To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.
- To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

6. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the

password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

An imported certificate is added to the `key.p12` file or the `root-key.p12` file.

If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

7. From the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**.

The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:

- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

- Linux UNIX `profile_root/properties/portdef.props`
- Windows `profile_root\properties\portdef.props`

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see the topic Updating ports in an existing profile in the WebSphere Application

Server Network Deployment information center. Run the `updatePorts.ant` file through the **ws_ant** script detailed in this topic.

| If you are installing | Next step |
|--|---|
| On a Linux or Windows platform, <i>and have root or Administrator group privileges</i> | The Linux or Windows Service Definition page is displayed. Proceed to step 8. |

8. In the Service Definition page, indicate whether or not to use a Windows service or Linux service to run WebSphere ESB

Windows The Windows Service Definition page displays for the Windows platform only when the ID that installs the Windows service has the Administrator group privilege. If the profile is configured as a Windows service, the product starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Windows service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

Important: If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, it must belong to the Administrator group, and it must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

IPv6 considerations when running profiles as Windows services

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the WebSphere ESB Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the WebSphere ESB Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

Linux The Linux Service Definition page is displayed only if the current operating system is a supported version of Linux and the current user has the appropriate permissions.

WebSphere ESB attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Linux service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

By default, WebSphere ESB is not selected to run as a Linux service.

To create the service, the user who runs the Profile Management Tool must be the root user. If you run the Profile Management Tool with a non-root user ID, the Linux Service Definition page is not displayed, and no service is created.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have proper privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

9. Configure the databases using a design file.
 - a. Select **Use a database design file**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

- e. Click **Next**.

If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration. For more information on using a design file for database configuration, see “Creating database design files by using the database design tool” on page 97.

10. If you chose not to configure the databases using a design file, specify the database details on the database configuration panels.
 - a. On the Database Configuration page, perform the following actions:
 - From the **Select a database product** drop-down, select the database product to be used by the profile.
 - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

If the database product you are using with the software already exists, select **Use an existing local or remote database**.
 - Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

If you do not select the checkbox, the scripts are output to the default directory.
 - Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.
 - b. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. For information about each required field, see Database configuration fields for Profile Management Tool configuration. You can configure parameters for the following databases:

DB2 After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

DB2 Universal Database for z/OS

You cannot create a new database using DB2 Universal Database for z/OS. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

Microsoft SQL Server

After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Database Configuration - Part 3 page. When you have completed configuration on the Database Configuration - Part 3 page, click **Next** to go to the Profile Summary page.

Oracle You cannot create a new database using this database. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Database Configuration - Part 3 page. When you have completed configuration on the Database Configuration - Part 3 page, click **Next** to go to the Profile Summary page.

DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox)

After setting the values for the DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox) database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

11. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

12. Manually configure the SMTP server to enable mail notifications. Refer to Configuring the SMTP server.

Results

You have created the deployment manager profile for the network deployment configuration.

What to do next

Create and configure one or more custom profiles (managed nodes) for your network deployment configuration.

Creating custom profiles (managed nodes) using the Profile Management Tool

You can use the Profile Management Tool to create and configure custom profiles (managed nodes) for your network deployment configuration.

Before you begin

To use the node (custom profile), you must federate the node to an existing deployment manager. If you want to federate the node during the creation process, the deployment manager must be running.

Tip: Instead of using the Profile Management Tool, you can use the `manageprofiles` command-line utility to create profiles, as described in *Configuring the software using command-line utilities and wsadmin*.

About this task

You can choose to federate the node (custom profile) to an existing deployment manager during the creation process, or federate the node later using the `addNode` command. If you decide to federate the node during the creation process, the tool sets the Common database configuration to the same database as the deployment manager. If you decide not to federate the node, the database configuration is left unconfigured.

Procedure

1. Start the Profile Management Tool.

Use one of the following commands:

- **Linux** **UNIX** `install_root/bin/ProfileManagement/pmt.sh`
- **Windows** `install_root\bin\ProfileManagement\pmt.bat`

The Welcome page displays.

2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

The **Profiles** tab displays.

The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.

3. In the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.

4. On the Environment Selection page, select the profile to be created.
 - a. Select **WebSphere Enterprise Service Bus > Enterprise service bus custom profile**
 - b. Click **Next**

The Profile Creation Options page displays.

5. On the Profile Creation Options page, decide whether to create the custom profile using the **Typical profile creation** or **Advanced profile creation** option.

Table 47. Selecting the profile creation option for your custom profile

| Select | When you want to . . . |
|----------------------------------|--|
| Typical profile creation | <p>Let the Profile Management Tool</p> <ul style="list-style-type: none"> Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell. Install the administrative console. Create a personal security certificate for the profile. The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year. Create a root signing security certificate for signing other certificates. The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years. Create a system service to run the server. Applicable only when your operating system and the privileges of your user account permit. Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database. Select any of the supported database products and set the database configuration for the CommonDB. |
| Advanced profile creation | <ul style="list-style-type: none"> Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable). Deploy the administrative console. Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets). Create a web server definition. Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services. |

If you selected **Typical profile creation**, go to step 6.

If you selected **Advanced profile creation** , go to step 9 on page 142.

- In the **Federation** page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Check **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a WebSphere ESB deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.

- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

Click **Next** to continue.

7. On the Database Configuration page, select the database used by the deployment manager and specify the location of the JDBC driver classpath files.
8. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.

You have completed the steps to create the profile with default configuration settings.

The remaining steps in this topic are for the **Advanced profile creation**.

9. In the Profile Name and Location page, perform the following steps:
 - a. In the **Profile name** field, specify a unique name or accept the default value.

Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Profile commands in a multiprofile environment for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

- b. In the **Profile directory** field, enter the directory for the profile or use the **Browse...** button to go to the profile directory.

The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform:

- **Linux** **UNIX** `install_root/profiles/profile_name`
- **Windows** `install_root\profiles\profile_name`

where *profile_name* is the name you specified. An error message is displayed if:

- The *profile_name* you specify is not unique.
 - The directory you specify is not empty.
 - Your user ID does not have sufficient permissions for the directory.
 - There is not sufficient space to create the profile.
- c. Optional: Select the **Make this profile the default** check box if you wish to make the profile you are creating the default profile.

When a profile is made to be the default profile, commands work automatically with it.

Note: This check box appears only if you have an existing profile on your system.

The first profile that you create on a workstation is the default profile.

The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See *Profile commands in a multiprofile environment* for more information.

- d. Click **Next**.

Note: If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

10. In the **Node and Host Names** page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value.

Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.

- In the **Host name** field, enter a name for the host or accept the default value.

Click **Next** to display the Federation page.

11. In the **Federation** page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Check **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a WebSphere ESB deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

Click **Next** to continue.

12. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

Perform the following actions:

- To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.
- To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:
 - In the **Path** field, enter the directory path to the existing certificate.
 - In the **Password** field, enter the password for the certificate
 - In the **Keystore type** field, select the keystore type for the certificate you are importing.
 - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
 - Click **Next** to display the Security Certificate (Part 2) page

When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the trust.p12 file.

13. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the `deleteKeyStore` task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

An imported certificate is added to the `key.p12` file or the `root-key.p12` file.

If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. In the Database Configuration page, select a database product and the location of the JDBC drivers.
15. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

16. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.
17. Manually configure the SMTP server to enable mail notifications. Refer to *Configuring the SMTP server*.

Results

You have created your custom profile.

What to do next

The node within the profile is empty until you federate the node to a deployment manager, then use the administrative console to customize it.

In a network deployment environment, you must create and configure databases, create other custom profiles and federate them to your deployment manager, create servers, create clusters if you want workload management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

Federating custom nodes to a deployment manager

You can use the **addNode** command to federate a custom node into a deployment manager cell. The following instructions guide you through the process of federating and deploying custom nodes.

Before you begin

Before using this procedure, ensure that the following prerequisites are met:

- You have installed WebSphere ESB and created an IBM Business Process Manager or WebSphere ESB deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:
 - Linux UNIX `profile_root/bin/startManager.sh`
 - Windows `profile_root\bin\startManager.bat`
- The deployment manager has been augmented into an IBM Business Process Manager or WebSphere ESB deployment manager.
- The deployment manager is at the same release level or higher than the custom profile you created or augmented.
- The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
- You do not plan to use this custom node as a migration target.

About this task

Federate a custom node so that it can be managed by a deployment manager. Use the **addNode** command to federate a custom profile into a deployment manager cell.

Procedure

1. Go to the bin directory of the custom profile you want to federate. Open a command window and go to one of the following directories (from a command line), depending on platform (where *profile_root* represents the installation location of the custom profile):

- **Linux** **UNIX** *profile_root/bin*
- **Windows** *profile_root\bin*

2. Issue the **addNode** command.

Issue one of the following commands from the command line if security is not enabled:

- **Linux** **UNIX** *./addNode.sh deployment_manager_host deployment_manager_SOAP_port*
- **Windows** *addNode.bat deployment_manager_host deployment_manager_SOAP_port*

Issue one of the following commands from the command line if security is enabled:

- **Linux** **UNIX** *./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username userID_for_authentication -password password_for_authentication*
- **Windows** *addNode.bat deployment_manager_host deployment_manager_SOAP_port -username userID_for_authentication -password password_for_authentication*

An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

ADMU0003I: Node DMNDID2Node03 has been successfully federated.

Results

The custom profile is federated into the deployment manager. For more information about the **addNode** command and its parameters, see the topic Using wsadmin scripting to run the addNode command in the WebSphere Application Server Network Deployment information center.

What to do next

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Federating stand-alone server profiles to a deployment manager

Learn how to use the **addNode** command to federate a stand-alone server profile into a deployment manager cell. After federation, a node agent process is created. Both this node agent and the server process are managed by the deployment manager. If you federate a stand-alone server profile and include all of its applications, the act of federation installs the applications on the deployment manager. A stand-alone server profile can be federated only if there are no other federated profiles.

Before you begin

Ensure that the following prerequisites are met:

- You have created a WebSphere ESB deployment manager.
- The deployment manager has been augmented into a WebSphere ESB deployment manager.
- The stand-alone server profile is a WebSphere ESB profile.
- The stand-alone server profile does not use file store or DB2 Express data store for its messaging engines. If you created the profile using the **Typical** option in the Profile Management Tool, the profile uses these options. You cannot federate it to a deployment manager.
- The stand-alone server uses a database driver that supports remote access, such as DB2 or Java toolbox JDBC.
- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:
 - **Linux** **UNIX** `profile_root/bin/startManager.sh`
 - **Windows** `profile_root\bin\startManager.bat`
- The stand-alone server is *not* running. If it is, stop it either by selecting **Stop the server** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the stand-alone server profile:
 - **Linux** **UNIX** `profile_root/bin/stopServer.sh`
 - **Windows** `profile_root\bin\stopServer.bat`
- The deployment manager is at the same release level or higher than the profile you created or augmented.
- The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
- No other nodes are federated to the deployment manager.

If you federate a stand-alone server profile when the deployment manager is not running or is not available for other reasons, profile federation fails and the resulting profile is unusable. You must then move this stand-alone server profile directory out of the profile repository before creating another profile with the same profile name.

About this task

Perform this task when you have an existing stand-alone server profile and you need to add the capabilities that network deployment offers to that server (central management or clustering). This function provides a growth path for an existing stand-alone server profile. However, you are limited to a single cluster configuration for this deployment environment. For a description of the single cluster pattern, see Single cluster topology.

Perform this task once for each cell and only for the first profile federated to the cell. Do not perform this task if the cell already has federated nodes. If do not have an existing stand-alone server profile, create the environment using custom profiles; for example, see Creating custom profiles (managed nodes), by using the Profile Management Tool.

Procedure

1. Go to the bin directory of the stand-alone server profile you want to federate. Open a command window and go to one of the following directories, depending on platform, where *profile_root* represents the installation location of the stand-alone server profile:

- **Linux** **UNIX** *profile_root/bin*
- **Windows** *profile_root\bin*

2. Issue the **addNode** command.

Issue one of following commands if security is not enabled. The port parameter is optional and can be omitted if you used the default port numbers when creating the deployment manager profile:

- **Linux** **UNIX** *./addNode.sh deployment_manager_host deployment_manager_SOAP_port -includeapps -includebuses*
- **Windows** *addNode.bat deployment_manager_host deployment_manager_SOAP_port -includeapps -includebuses*

Issue one of the following commands if security is enabled:

- **Linux** **UNIX** *./addNode.sh deployment_manager_host deployment_manager_SOAP_port -username userID_for_authentication -password password_for_authentication -localusername localuserID_for_authentication -localpassword localpassword_for_authentication -includeapps -includebuses*
- **Windows** *addNode.bat deployment_manager_host deployment_manager_SOAP_port -username userID_for_authentication -password password_for_authentication -localusername localuserID_for_authentication -localpassword localpassword_for_authentication -includeapps -includebuses*

An output window opens. If you see a message like the following one, your stand-alone server profile was federated successfully:

ADMU0003I: Node DMNDID2Node02 has been successfully federated.

Results

The stand-alone server profile is federated into the deployment manager. For more information about the **addNode** command and its parameters, see the topic Using wsadmin scripting to run the addNode command in the WebSphere Application Server Network Deployment information center.

Deleting profiles using the manageprofiles command-line utility

You can delete a profile from the command line using the **manageprofiles** command-line utility.

Before you begin

For more information about the **manageprofiles** command-line utility, see manageprofiles command-line utility.

Procedure

1. Open a command prompt and run one of the following commands, based on your operating system:

- **Linux** **UNIX** *manageprofiles.sh -delete -profileName profile_name*
- **Windows** *manageprofiles.bat -delete -profileName profile_name*

The variable *profile_name* represents the name of the profile that you want to delete.

2. Confirm that the profile deletion has completed by checking the following log file:

- **Linux** **UNIX** `install_root/logs/manageprofiles/
profile_name_delete.log`
- **Windows** `install_root\logs\manageprofiles\profile_name_delete.log`

What to do next

If you plan to recreate a deleted profile using both the same profile name and the same database names that are associated with the deleted profile, you must manually delete the associated database names before you attempt to recreate the profile name and the database names.

Configuring the environment using manageprofiles and wsadmin

You can achieve the same configurations that you set up using the Profile Management Tool and deployment environment wizard by using the **manageprofiles** command-line utility and the **wsadmin** command.

Creating profiles using the manageprofiles command-line utility:

You can create a profile from the command line using the **manageprofiles** command-line utility and a property file.

Before you begin

Before you run the **manageprofiles** command-line utility ensure that you have completed the following tasks:

- You have reviewed the full list of prerequisites for creating or augmenting a profile at Prerequisites for creating or augmenting profiles.
- You have reviewed example profile creation commands .
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Security role required for this task: See Granting write permission of files and directories to nonroot users for profile creation.

To use the **manageprofiles** command-line utility to create a profile, perform the following steps.

Procedure

1. Determine the kind of profile you want to create, which in turn determines the template to use for your new profile (using the **-templatePath** option). The following templates are available:
 - `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile, which defines a stand-alone server.
 - `dmgr.esbserver`: for a WebSphere Enterprise Service Bus deployment manager profile, which defines a deployment manager.
 - `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile, which, when federated to a deployment manager, defines a managed node. Do not federate a node unless the deployment manager you are federating to

is at a release level the same or higher than that of the custom profile you are creating. WebSphere Enterprise Service Bus profiles can use a WebSphere Enterprise Service Bus or WebSphere Process Server deployment manager.

Templates for each profile are located in the *install_root/profileTemplates* directory.

2. Determine which parameters are required for your type of profile by reviewing the example profile creation commands in *manageprofile* examples.
3. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** command-line utility parameters topic to see if they are what you need for your profile.

Note: If you create profiles in WebSphere ESB using the **manageprofiles** command-line utility without specifying the *samplesPassword* parameter, the *INSTCONFPARTIALSUCCESS* message is returned. This occurs when the following criteria are met:

- You installed the samples during WebSphere ESB or WebSphere Application Server installation.
 - You use the **manageprofiles** command-line utility to create the profiles.
 - The *samplesPassword* parameter is not specified in the **manageprofiles** command-line utility.
4. Run the file from the command line. Here are some simple examples. For more complex examples, see *manageprofiles* examples. Linux UNIX

- Linux UNIX `manageprofiles.sh -create -templatePath install_root/profileTemplates/default.esbserver`

Windows

- Windows `manageprofiles.bat -create -templatePath install_root\profileTemplates\default.esbserver`

If you have created a response file, use the **-response** parameter: `-response myResponseFile`

The following example shows a response file for a create operation:

```
create
profileName=testResponseFileCreate
profilePath=profile_root
templatePath=install_root/profileTemplates/default.esbserver
nodeName=myNodeName
cellName=myCellName
hostName=myHostName
omitAction=myOptionalAction1, myOptionalAction2
```

The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

5. Manually configure the SMTP server to enable mail notifications. See *Configuring the SMTP server*.

What to do next

You can see that your profile creation completed successfully if you receive a *INSTCONFSUCCESS: Profile creation succeeded.* message, and you can check the following log file: Linux UNIX Windows

- **Linux** **UNIX** `install_root/logs/manageprofiles/profile_name_create.log`
- **Windows** `install_root\logs\manageprofiles\profile_name_create.log`

Run the Installation Verification Test (IVT) tool to verify that the profile was created successfully. To do this, run the following command:

- **Linux** **UNIX** `profile_root/bin/wbi_ivt.sh`
- **Windows** `profile_root\bin\wbi_ivt.bat`

Related concepts:

“JDBC drivers and locations” on page 40

The following tables list the supported JDBC drivers. The first table contains the names and locations of the JDBC drivers that are provided with the product. The second table contains the names of the JDBC drivers that are supported but not provided with the product.

Creating deployment manager and custom profiles using manageprofiles after installation:

After performing an installation you can create deployment manager and custom (managed node) profiles using the Profile Management Tool or the manageprofiles command-line utility.

The information in this section describes how to use the manageprofiles command-line utility to create deployment manager and custom (managed node) profiles for a network deployment configuration. It assumes that you have run the installer and have performed an installation.

For information about using the Profile Management Tool to create deployment manager and custom (managed node) profiles after performing an installation, see Creating custom profiles (managed nodes) by using the Profile Management Tool.

Creating deployment manager and custom profiles using manageprofiles:

Use the **manageprofiles** command-line utility to create deployment manager and custom profiles for a network deployment configuration.

Before you begin

Before you run the **manageprofiles** command-line utility ensure that you have completed the following tasks:

- You have reviewed the full list of prerequisites for creating or augmenting a profile
- You have reviewed example profile creation commands
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

About this task

This task describes how to use the **manageprofiles** command-line utility to create deployment manager and custom profiles for a network deployment configuration.

To use the **manageprofiles** command-line utility to create the profiles, perform the following steps.

Procedure

1. Determine the kind of profile you want to create, which in turn determines the template to use for your new profile (using the **-templatePath** option). The following templates are available:
 - **dmgr.esbserver**: for a WebSphere Enterprise Service Bus deployment manager profile, which defines a deployment manager.
 - **managed.esbserver**: for a WebSphere Enterprise Service Bus custom profile, which, when federated to a deployment manager, defines a managed node. Do not federate a node unless the deployment manager you are federating to is at a release level the same or higher than that of the custom profile you are creating. WebSphere Enterprise Service Bus profiles can use a Enterprise Service Bus or WebSphere ESB Process Server deployment manager.

Templates for each profile are located in the *install_root/profileTemplates* directory.

2. Determine which parameters are required for your type of profile by reviewing the example profile creation commands.
3. Determine the values that you want to supply for the profile by reviewing the default values in the **manageprofiles** topic to see if they are what you need for your profile.
4. Run the file from the command line. For example:

- `manageprofiles -create -templatePath install_root/profileTemplates/dmgr.esbserver`

If you have created a response file, specify the **-response** parameter without any other parameters. For example:

```
manageprofiles -response myResponseFile
```

The following example shows a response file for a create operation:

```
create
profileName=testResponseFileCreate
profilePath=profile_root

templatePath=install_root/profileTemplates/dmgr.esbserver
nodeName=myNodeName
cellName=myCellName
hostName=myHostName
omitAction=myOptionalAction1, myOptionalAction2
```

The status is written to the console window when the command is finished running. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

For more complex examples, see the following:

- Examples: Creating profiles with **manageprofiles** command-line utility using a DB2 database.
- Examples: Creating profiles with **manageprofiles** command-line utility using an Oracle database
- **manageprofiles** parameters for Common database configuration (per database product)
- **manageprofiles** parameters for Common Event Infrastructure database configuration (per database product)

5. Manually configure the SMTP server to enable mail notifications. Refer to Configuring the SMTP server.

manageprofiles examples:

The examples in this section show how to create stand-alone, deployment manager, and custom (managed node) profiles using the **manageprofiles** command-line utility.

*Examples: Creating WebSphere ESB profiles with **manageprofiles** command-line utility using an Oracle database:*

This topic contains example profile creation commands to help you create stand-alone server, deployment manager, and custom profiles using the **manageprofiles** command-line utility on your installation with an Oracle database.

Stand-alone server profile

The following command example creates an WebSphere ESB stand-alone profile called *my_WESBSA_profile* on a Windows server.

The parameters in Specified **manageprofiles** command-line utility parameters and Defaulted **manageprofiles** command-line utility parameters specify the following features:

- The Oracle database product will be used for the Common database, which is assumed to already exist on the localhost. The database is set to be configured later (the **-dbDelayConfig "true"** command parameter value specifies that configuration scripts be created but not run). For complete listings of database-related **manageprofiles** parameters, see the topic *manageprofiles parameters*.
- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

- Administrative security will be enabled.

All user IDs specified for profile creation should already exist in the database before any database configuration is performed:

- If **dbDelayConfig** is set to false, the user IDs must be created before profile creation.
- If **dbDelayConfig** is set to true, the user IDs can be created at the same time as the database tables after profile creation is complete and before starting the server.

Table 48 shows **manageprofiles** command-line utility parameters with example values used to create a stand-alone server profile.

*Table 48. Specified **manageprofiles** command-line utility parameters*

| Parameter | Value |
|-----------|-------|
| -create | N/A |




Table 48. Specified **manageprofiles** command-line utility parameters (continued)

| Parameter | Value |
|----------------------|---|
| -templatePath | "install_root\profileTemplates\default.esbserver" (must be fully qualified) |
| -profileName | "my_WESBSA_profile" |
| -enableAdminSecurity | "true" |
| -adminPassword | "admin_pwd" |
| -adminUserName | "admin_id" |
| -dbServerPort | "1521" |
| -dbType | "ORACLE" |
| -dbName | "CMNDB" |
| -dbDelayConfig | "true" |
| -dbPassword | "db_pwd" |
| -configureBSpace | "true" |
| -configureBRM | "false" |
| -samplesPassword | "samples_pwd" |

Tip: The **samplesPassword** parameter is only required when using Samples.

Table 49 shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 49. Defaulted **manageprofiles** command-line utility parameters

| Parameter | Default values |
|--|---|
| -profilePath | "install_root\profiles\my_WESBSA_profile" |
| -hostName | "host_name" |
| -nodeName | "host_nameNodenode_number" |
| -cellName | "host_nameNodenode_numbercell_numberCell" |
|  -winserviceCheck | "true" |
|  -winserviceAccountType | "localsystem" |
|  -winserviceUserName | "Administrator" |
| -dbOutputScriptDir | "install_root\profiles\my_WESBSA_profile\dbscripts" |
| -dbJDBCClasspath | "install_root\jdbcDrivers\Oracle" |

Deployment manager profile

The following command example creates a deployment manager profile called *my_WESBDMGR_profile* on a Windows server.

The parameters in Table 50 on page 156, Table 51 on page 156, and Table 52 on page 157 specify the following:

- The Oracle database product will be used for the Common database, which is set to be created and configured on the localhost during the profile creation process.

- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

- Administrative security will be enabled.

WebSphere Enterprise Service Bus example

Specified **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a deployment manager profile.

Table 50. Specified **manageprofiles** command-line utility parameters

| Parameter | Value |
|----------------------|--|
| -create | N/A |
| -templatePath | "install_root\profileTemplates\dmgr.esbserver" (must be fully qualified) |
| -profileName | "my_WESBDMGR_profile" |
| -enableAdminSecurity | "true" |
| -adminPassword | "admin_pwd" |
| -adminUserName | "admin_id" |
| -dbType | "ORACLE" |
| -dbName | "CMNDB" |
| -dbDelayConfig | "true" |
| -dbPassword | "db_pwd" |
| -dbHostName | "remote_host_name" |
| -dbServerPort | "1521" |

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 51. Defaulted **manageprofiles** command-line utility parameters





| Parameter | Default values |
|--|---|
| -profilePath | "install_root\profiles\my_WESBDMGR_profile" |
| -hostName | "host_name" |
| -nodeName | "host_nameCellManagernode_number" |
| -cellName | "host_nameCellcell_number" |
|  -winserviceCheck | "true" |
|  -winserviceAccountType | "localsystem" |
|  -winserviceStartupType | "manual" |
|  -winserviceUserName | "Administrator" |

Table 51. Defaulted **manageprofiles** command-line utility parameters (continued)

| Parameter | Default values |
|--------------------|---|
| -dbJDBCClasspath | "install_root\jdbcDrivers\Oracle" |
| -dbOutputScriptDir | "install_root\profiles\my_WESBDMGR_profile\dbscripts" |

Additional **manageprofiles** command-line utility parameters for Oracle shows additional **manageprofiles** command-line utility parameters that are not displayed via the Profile Management Tool that can be specified to select your own user name and password combinations for Oracle.

Table 52. Additional **manageprofiles** command-line utility parameters for Oracle

| Parameter | Default values |
|-------------------|---|
| -dbCommonUserId | "common_db_userID" (used to create Common DB objects) |
| -dbCommonPassword | "common_db_pwd" |

Custom profile

The following command example creates a custom profile called *my_WESBCUSTOM_profile* on a Windows server.

This example is set to operate with the deployment manager profile created above.

The parameters in Table 53 and Table 54 on page 158 specify the following:

- The Oracle database product will be used for the Common database, which is assumed to already exist. The custom profile creation needs to point to the database used by the deployment manager to which the custom profile will be federated.
- Administrative security will be enabled on the deployment manager to which the custom profile will be federated.

See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

WebSphere Enterprise Service Bus example

Specified **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a custom profile.

Table 53. Specified **manageprofiles** command-line utility parameters

| Parameter | Value |
|---------------|---|
| -create | N/A |
| -templatePath | "install_root\profileTemplates\managed.esbserver" (must be fully qualified) |
| -profileName | "my_WESBCUSTOM_profile" |
| -dmgrHost | "remote_host" |

Table 53. Specified **manageprofiles** command-line utility parameters (continued)

| Parameter | Value |
|--------------------|--|
| -dmgrPort | "8879" (To find the -dmgrPort value, go to the <i>dmgr_profile_root</i> \logs directory for the deployment manager associated with this custom profile. In this directory, open the AboutThisProfile.txt file and find the value for the entry "Management SOAP connector port:".) |
| -dmgrAdminPassword | "admin_pwd" |
| -dmgrAdminUserName | "admin_id" |
| -federateLaterWESB | "false" |
| -dbType | "ORACLE" |
| -dbJDBCClasspath | "install_root\jdbcDrivers\Oracle" |

Remember: If the **federateLaterWESB** parameter is set to true, then the dmgrHost, dmgrPort, dmgrAdminPassword and dmgrAdminUserName do not need to be specified.

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 54. Defaulted **manageprofiles** command-line utility parameters

| Parameter | Default values |
|--------------|---|
| -profilePath | "install_root\profiles\my_WESBCUSTOM_profile" |
| -hostName | "host_name" |
| -nodeName | "host_nameNodenode_number" |

manageprofiles parameters for Common database configuration (per database product):

You use specific **manageprofiles** command-line utility parameters to configure the Common database. Parameters you specify can differ depending on the database product you are using and on the type of profile you are creating.

The tables in this topic show the **manageprofiles** parameters available to configure the Common database using any supported database product. Parameters associated with Common database configuration generally have a "-db" prefix; for example **-dbType**, and **-dbDelayConfig**. Also shown are the equivalent field names for the parameters as they appear in the Profile Management Tool.

For a complete list of **manageprofiles** parameters, including default values, see the topic **manageprofiles** parameters. For examples of using **manageprofiles** commands to create or augment various types of profiles, see "Examples: Creating WebSphere ESB profiles with **manageprofiles** command-line utility using a DB2 database" on page 170.

To view available parameters for database configuration, choose your database product from the following list:

- "On DB2 Universal" on page 159
- "On DB2 Data Server" on page 160
- "On DB2 for IBM i (Toolbox) and DB2 for i5/OS (Toolbox)" on page 160

- “On DB2 for z/OS v8 and DB2 for z/OS v9” on page 161
- “On Oracle” on page 162
- “On Microsoft SQL Server” on page 163

Note that only the **-dbType** and **-dbJDBCClasspath** parameters are available for custom profiles. This is because you are simply identifying the type and driver location for the Common database used by the deployment manager to which you will federate the custom profile.

On DB2 Universal

Table 55 shows the **manageprofiles** parameters available to configure the Common database used by a stand-alone server, deployment manager, or custom profile on DB2 Universal.

Table 55. Available manageprofiles parameters for configuration of Common database using DB2 Universal

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| For custom profiles | |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbType | Choose the database product used on the deployment manager |
| For stand-alone server or deployment manager profiles | |
| -cdbSchemaName A new parameter that take precedence over dbSchemaName if both are specified.-dbSchemaName Note: Deprecated in V7. | Schema name |
| -dbCommonForME (for stand-alone server profiles only) | Use this database for Messaging Engines (MEs) |
| -dbCreateNew | N/A |
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbDriverType | N/A |
| -dbHostName | Database server host name (for example IP address) |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbName | Common database name |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -dbPassword | Password for database authentication |
| -dbServerPort | Server port |
| -dbType | Choose a database product |
| -dbUserId | User name to authenticate with the database |
| -fileStoreForME (for stand-alone server profiles only) | Use a file store for Messaging Engines (MEs) |
| N/A | Override the destination directory for generated scripts |

On DB2 Data Server

Table 56 shows the **manageprofiles** parameters available to configure the Common database used by a stand-alone server, deployment manager, or custom profile on DB2 Universal.

Table 56. Available *manageprofiles* parameters for configuration of Common database using DB2 Data Server

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| For custom profiles | |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbType | Choose the database product used on the deployment manager |
| For stand-alone server or deployment manager profiles | |
| -cdbSchemaName A new parameter that take precedence over dbSchemaName if both are specified.-dbSchemaName Note: Deprecated in V7. | Schema name |
| -dbCommonForME (for stand-alone server profiles only) | Use this database for Messaging Engines (MEs) |
| -dbCreateNew | N/A |
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName | Database server host name (for example IP address) |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbName | Common database name |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -dbPassword | Password for database authentication |
| -dbServerPort | Server port |
| -dbType | Choose a database product |
| -dbUserId | User name to authenticate with the database |
| -fileStoreForME (for stand-alone server profiles only) | Use a file store for Messaging Engines (MEs) |
| N/A | Override the destination directory for generated scripts |

On DB2 for IBM i (Toolbox) and DB2 for i5/OS (Toolbox)

Table 57 on page 161 shows the **manageprofiles** parameters available to configure the Common database used by a stand-alone server, deployment manager, or custom profile on a database supplied with an i5/OS or IBM i operating system.

Table 57. Available *manageprofiles* parameters for configuration of Common database using a database supplied with an i5/OS or IBM i operating system

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| For custom profiles | |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbType | Choose the database product used on the deployment manager |
| For stand-alone server or deployment manager profiles | |
| -dbCommonForME (for stand-alone server profiles only) | Use this database for Messaging Engines (MEs) |
| -dbCreateNew (must always be true) | N/A |
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName (for Toolbox driver, you need to specify the remote database host name) | Database server host name (for example IP address) |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbName | Common database name |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -dbPassword | Password for database authentication |
| -cdbSchemaName | Database collection name |
| A new parameter that take precedence over dbSchemaName if both are specified.-dbSchemaName Note: Deprecated in V7. | |
| -dbType | Choose a database product |
| -dbUserId | User name to authenticate with the database |
| -fileStoreForME (for stand-alone server profiles only) | Use a file store for Messaging Engines (MEs) |
| N/A | Override the destination directory for generated scripts |

On DB2 for z/OS v8 and DB2 for z/OS v9

Table 58 shows the **manageprofiles** parameters available to configure the Common database used by a stand-alone server, deployment manager, or custom profile on DB2 for z/OS v8 or DB2 for z/OS v9.

Table 58. Available *manageprofiles* parameters for configuration of Common database using DB2 for z/OS v8 or DB2 for z/OS v9

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|----------------------------|--|
| For custom profiles | |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |

Table 58. Available manageprofiles parameters for configuration of Common database using DB2 for z/OS v8 or DB2 for z/OS v9 (continued)

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| -dbType | Choose the database product used on the deployment manager |
| For stand-alone server or deployment manager profiles | |
| -dbCommonForME (for stand-alone server profiles only) | Use this database for Messaging Engines (MEs) |
| -dbConnectionLocation | Connection location |
| -dbCreateNew (must always be false) | N/A |
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName | Database server host name (for example IP address) |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbName | Common database name |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -dbPassword | Password for database authentication |
| -cdbSchemaName A new parameter that take precedence over dbSchemaName if both are specified.-dbSchemaName Note: Deprecated in V7. | Database alias name |
| -dbServerPort | Server port |
| -dbStorageGroup | Storage group name |
| -dbType | Choose a database product |
| -dbUserId | User name to authenticate with the database |
| -fileStoreForME (for stand-alone server profiles only) | Use a file store for Messaging Engines (MEs) |
| N/A | Override the destination directory for generated scripts |

On Oracle

Table 59 shows the **manageprofiles** parameters available to configure the Common database used by a stand-alone server, deployment manager, or custom profile on Oracle.

Table 59. Available manageprofiles parameters for configuration of Common database using Oracle

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|----------------------------|--|
| For custom profiles | |

Table 59. Available `manageprofiles` parameters for configuration of Common database using Oracle (continued)

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files You must install the <code>ojdbc6.jar</code> driver to access the Oracle database. Note: Oracle 10g does not contain the <code>ojdbc6.jar</code> driver. You can download it from the Oracle web site. |
| -dbType | Choose the database product used on the deployment manager |
| | |
| For stand-alone server or deployment manager profiles | |
| -dbCommonForME (for stand-alone server profiles only) | Use this database for Messaging Engines (MEs) |
| -dbCreateNew (must always be false) | N/A |
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbDriverType | JDBC driver type |
| -dbHostName | Database server host name (for example IP address) |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbName | Common database name |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -dbPassword | Common database password |
| -dbServerPort | Server port |
| -dbType | Choose a database product |
| -dbUserId | Common database user name |
| -fileStoreForME (for stand-alone server profiles only) | Use a file store for Messaging Engines (MEs) |
| -dbLocation (required only if -dbDelayConfig is set to true) | Directory of database server installation |
| -dbSysPassword | Password |
| -dbSysUserId | System administrator user name |
| N/A | Override the destination directory for generated scripts |

On Microsoft SQL Server

Table 60 on page 164 shows the **manageprofiles** parameters that are available to configure the Common database that is used by a stand-alone server, deployment manager, or custom profile on Microsoft SQL Server. The following JDBC drivers are available for this database:

- Microsoft SQL Server JDBC Driver, version 1.2
- Microsoft SQL Server JDBC Driver, version 2.0

If you plan to use Microsoft SQL Server 2005 or 2008 with a standalone profile, and will put the messaging engine tables in the Common Database, then you must perform the following steps:

1. Manually add four schemas to the Common database before creating stand-alone server profiles. These schemas are XXXSS00, XXXSA00, XXXCM00, and XXXBM00, where XXX is the first three characters of the name of the Common database.
2. Pass the dbCommonForME=true parameter during profile creation. The following command configures the Messaging Engines on SQL Server with the schemas that were defined above. The command uses the dbUserId and dbPassword that you specified for CommonDB.

For Microsoft SQL Server JDBC 1.2 driver

For Microsoft SQL Server JDBC 2.0 driver

Table 60. Available manageprofiles parameters for configuration of Common database using Microsoft SQL Server

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| For custom profiles | |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbType | Choose the database product used on the deployment manager |
| For stand-alone server or deployment manager profiles | |
| -dbCommonForME (for stand-alone server profiles only) | Use this database for Messaging Engines (MEs) |
| -dbCreateNew | N/A |
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbDriverVersion | JDBC driver version |
| -dbHostName | Database server host name (for example IP address) |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbName | Common database name |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -dbPassword | Password for database authentication |
| -dbServerPort | Server port |
| -dbType | Choose a database product |
| -dbUserId | Common database user name |
| -fileStoreForME (for stand-alone server profiles only) | Use a file store for Messaging Engines (MEs) |
| -ceiDbServerName | Database server name |
| -ceiSaPassword | Admin user password |
| -ceiSaUser | Admin user name |
| N/A | Override the destination directory for generated scripts |

manageprofiles parameters for Common Event Infrastructure database configuration (per database product):

You use specific **manageprofiles** command-line utility parameters to configure the Common Event Infrastructure database used by a stand-alone server profile. Parameters you specify can differ depending on the database product you are using.

The tables in this topic show the **manageprofiles** parameters available to configure the Common Event Infrastructure database using any supported database product. Also shown are the equivalent field names for the parameters as they appear in the Profile Management Tool. You configure the Common Event Infrastructure database using the **manageprofiles** command-line utility only for stand-alone server profiles. Configuration of this database for use by deployment manager profiles must be done through the administrative console or scripting. See the topic Configuring the event database for more information.

For a complete list of **manageprofiles** parameters, including default values, see the topic **manageprofiles** parameters.

To view available parameters for database configuration, choose your database product from the following list:

- “On DB2 Universal”
- “On DB2 Data Server” on page 166
- “On DB2 for IBM i (Toolbox) and DB2 for i5/OS (Toolbox)” on page 166
- “On DB2 for z/OS v8 and DB2 for z/OS v9” on page 167
- “On Oracle” on page 168
- “On Microsoft SQL Server” on page 169

On DB2 Universal

Table 61 shows the **manageprofiles** parameters available to configure the Common Event Infrastructure database used by a stand-alone server profile on DB2 Universal.

Table 61. Available manageprofiles parameters for configuration of Common Event Infrastructure database using DB2 Universal

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|------------------|--|
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName | Database server host name (for example IP address) |
| -ceiDbName | Common Event Infrastructure database name |
| -dbPassword | Password for database authentication |
| -dbServerPort | Server port |
| -dbType | Choose a database product |
| -dbUserId | User name to authenticate with the database |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |

Table 61. Available manageprofiles parameters for configuration of Common Event Infrastructure database using DB2 Universal (continued)

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|------------------------|--|
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -ceiOverrideDataSource | N/A (command-line only) |
| N/A | Override the destination directory for generated scripts |

On DB2 Data Server

Table 62 shows the **manageprofiles** parameters available to configure the Common Event Infrastructure database used by a stand-alone server profile on DB2 Data Server.

Table 62. Available manageprofiles parameters for configuration of Common Event Infrastructure database using On DB2 Data Server

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|------------------------|--|
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName | Database server host name (for example IP address) |
| -ceiDbName | Common Event Infrastructure database name |
| -dbPassword | Password for database authentication |
| -dbServerPort | Server port |
| -dbType | Choose a database product |
| -dbUserId | User name to authenticate with the database |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -ceiOverrideDataSource | N/A (command-line only) |
| N/A | Override the destination directory for generated scripts |

On DB2 for IBM i (Toolbox) and DB2 for i5/OS (Toolbox)

Table 63 on page 167 shows the **manageprofiles** parameters available to configure the Common Event Infrastructure database used by a stand-alone server profile on the database supplied with an i5/OS or IBM i operating system.

Table 63. Available *manageprofiles* parameters for configuration of Common Event Infrastructure database using a database supplied with an i5/OS or IBM i operating system

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| -ceiDbAlreadyConfigured | N/A (command-line only) |
| -ceiOverrideDataSource | N/A (command-line only) |
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName | Database server host name (for example IP address) |
| -ceiDbName | Common Event Infrastructure database name |
| -dbPassword | Password for database authentication |
| -dbType | Choose a database product |
| -dbUserId | User name to authenticate with the database |
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -cdbSchemaName | Database collection name |
| A new parameter that take precedence over dbSchemaName if both are specified.-dbSchemaName Note: Deprecated in V7. | |
| N/A | Override the destination directory for generated scripts |

On DB2 for z/OS v8 and DB2 for z/OS v9

Table 64 shows the **manageprofiles** parameters available to configure the Common Event Infrastructure database used by a stand-alone server profile on DB2 for z/OS v8 or DB2 for z/OS v9.

Table 64. Available *manageprofiles* parameters for configuration of Common Event Infrastructure database using DB2 for z/OS v8 or DB2 for z/OS v9

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|------------------------|--|
| -ceiBufferPool4k | N/A (command-line only) |
| -ceiBufferPool8k | N/A (command-line only) |
| -ceiBufferPool16k | N/A (command-line only) |
| -ceiDbName | Common Event Infrastructure database name |
| -ceiDiskSizeInMB | N/A (command-line only) |
| -ceiOverrideDataSource | N/A (command-line only) |
| -dbConnectionLocation | Connection location |
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName | Database server host name (for example IP address) |

Table 64. Available manageprofiles parameters for configuration of Common Event Infrastructure database using DB2 for z/OS v8 or DB2 for z/OS v9 (continued)

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -dbPassword | Password for database authentication |
| -cdbSchemaName A new parameter that take precedence over dbSchemaName if both are specified.-dbSchemaName Note: Deprecated in V7. | Database alias name |
| -dbStorageGroup | Storage group name |
| -dbType | Choose a database product |
| -dbUserId | User name to authenticate with the database |
| N/A | Override the destination directory for generated scripts |

On Oracle

Table 65 shows the **manageprofiles** parameters available to configure the Common Event Infrastructure database used by a stand-alone server profile on Oracle.

Table 65. Available manageprofiles parameters for configuration of Common Event Infrastructure database using Oracle

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName | Database server host name (for example IP address) |
| -ceiDbName | Common Event Infrastructure database name |
| -dbPassword | Password for database authentication |
| -dbServerPort | Server port |
| -dbType | Choose a database product |
| -dbSysPassword | Password |
| -dbSysUserId | System administrator user name |
| -dbUserId | Common database user name |
| -ceiInstancePrefix Note: Deprecated in 6.2 for all databases except Oracle and Microsoft SQL Server. | N/A (command-line only) |

Table 65. Available *manageprofiles* parameters for configuration of Common Event Infrastructure database using Oracle (continued)

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| -dbJDBCClasspath | Location (directory) of JDBC driver classpath files You must install the ojdbc6.jar driver to access the Oracle database. Note: Oracle 10g does not contain the ojdbc6.jar driver. You can download it from the Oracle web site. |
| -dbLocation (required only if -dbDelayConfig is set to true) | Directory of database server installation |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -ceiOverrideDataSource | N/A (command-line only) |
| N/A | Override the destination directory for generated scripts |

On Microsoft SQL Server

Table 66 shows the **manageprofiles** parameters that are available to configure the Common database that is used by a stand-alone server, deployment manager, or custom profile on Microsoft SQL Server. The following JDBC drivers are available for this database:

- Microsoft SQL Server JDBC Driver version 1.2
- Microsoft SQL Server JDBC Driver version 2.0

Table 66. Available *manageprofiles* parameters for configuration of Common Event Infrastructure database using Microsoft SQL Server.

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|---|--|
| -dbDelayConfig | Delay execution of database scripts (must select if using a remote database) |
| -dbHostName | Database server host name (for example IP address) |
| -ceiDbInstallDir (required only if -dbDelayConfig is set to true) | N/A (command-line only) |
| -ceiDbName | Common Event Infrastructure database name |
| -dbUserId | Common database user name |
| -dbPassword | Password for database authentication |
| -dbServerPort | Server port |
| -dbType | Choose a database product |
| -dbDriverVersion | JDBC driver version |
| -dbInstance (required only if -dbDelayConfig is set to true) | Instance name |

Table 66. Available *manageprofiles* parameters for configuration of Common Event Infrastructure database using Microsoft SQL Server. (continued)

| Parameter | Related field on Database Configuration pages in Profile Management Tool |
|--|--|
| -ceiDbUser Note: This user must be different from the dbUserId . Note: Deprecated in 6.2 for all databases except Microsoft SQL Server. | CEI database user name |
| -ceiDbPassword Note: Deprecated in 6.2 for all databases except Microsoft SQL Server. | CEI database password |
| -ceiInstancePrefix Note: Deprecated in 6.2 for all databases except Oracle and Microsoft SQL Server. | N/A (command-line only) |
| -dbOutputScriptDir | Database script output directory Note: Only available if Override the destination directory for generated scripts option is selected. The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup. |
| -ceiOverrideDataSource | N/A (command-line only) |
| -ceiSaPassword Note: Deprecated in 6.2 for all databases except Microsoft SQL Server. | Admin user password |
| -ceiSaUser Note: Deprecated in 6.2 for all databases except Microsoft SQL Server. | Admin user name |
| N/A | Override the destination directory for generated scripts |

*Examples: Creating WebSphere ESB profiles with *manageprofiles* command-line utility using a DB2 database:*

This topic contains example profile creation commands to help you create stand-alone server, deployment manager, and custom profiles using the **manageprofiles** command-line utility on your installation with a DB2 database.

Stand-alone server profile

The following command example creates an WebSphere ESB stand-alone server profile called *my_WESBSA_profile* on a Windows server. The parameters in Specified **manageprofiles** command-line utility parameters and Defaulted **manageprofiles** command-line utility parameters specify the following:

- The DB2 database product will be used for the Common database, which is assumed to already exist on the localhost. The database is set to be configured later (the **-dbDelayConfig "true"** command parameter value specifies that configuration scripts be created but not run). For complete listings of database-related **manageprofiles** parameters, see the topic *manageprofiles parameters*.
- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

- Administrative security will be enabled.

Specified **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a stand-alone server profile.

Table 67. Specified **manageprofiles** command-line utility parameters

| Parameter | Value |
|----------------------|---|
| -create | N/A |
| -templatePath | "install_root\profileTemplates\default.esbserver" (must be fully qualified) |
| -profileName | "my_WESBSA_profile" |
| -enableAdminSecurity | "true" |
| -adminPassword | "admin_pwd" |
| -adminUserName | "admin_id" |
| -dbServerPort | "50000" |
| -dbHostName | "localhost" |
| -dbType | "DB2_UNIVERSAL" or "DB2_DATASERVER" |
| -dbName | "CMNDB" |
| -dbCreateNew | "false" |
| -dbDelayConfig | "true" |
| -dbUserId | "db_id" |
| -dbPassword | "db_pwd" |
| -configureBSpace | "false" |
| -samplesPassword | "samples_pwd" |

Tip: The **samplesPassword** parameter is only required when using Samples.

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 68. Defaulted **manageprofiles** command-line utility parameters





| Parameter | Default values |
|--|---|
| -profilePath | "install_root\profiles\my_WESBSA_profile" |
| -hostName | "host_name" |
| -nodeName | "host_nameNodenode_number" |
| -cellName | "host_nameNodenode_numbercell_numberCell" |
|  -winserviceCheck | "true" |
|  -winserviceAccountType | "localsystem" |
|  -winserviceStartupType | "manual" |
|  -winserviceUserName | "Administrator" |

Table 68. Defaulted **manageprofiles** command-line utility parameters (continued)

| Parameter | Default values |
|--------------------|---|
| -dbOutputscriptDir | "install_root\profiles\my_WESBSA_profile\dbscripts" |
| -dbJDBCClasspath | "install_root\jdbcdrivers\DB2" |

Deployment manager profile

The following command example creates a deployment manager profile called *my_WESBDMGR_profile* on a Windows server.

The parameters in Table 69 and Table 70 on page 173 specify the following:

- The DB2 database product will be used for the Common database, which is set to be created and configured on the localhost during the profile creation process.
- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

- Administrative security will be enabled.

WebSphere Enterprise Service Bus example





Specified **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a deployment manager profile.

Table 69. Specified **manageprofiles** command-line utility parameters

| Parameter | Value |
|----------------------|--|
| -create | N/A |
| -templatePath | "install_root\profileTemplates\dmgr.esbserver" (must be fully qualified) |
| -profileName | "my_WESBDMGR_profile" |
| -enableAdminSecurity | "true" |
| -adminPassword | "admin_pwd" |
| -adminUserName | "admin_id" |
| -dbType | "DB2_UNIVERSAL" or "DB2_DATASERVER" |
| -dbName | "CMNDB" |
| -dbCreateNew | "true" |
| -dbDelayConfig | "false" |
| -dbUserId | "db_id" |
| -dbPassword | "db_pwd" |
| -dbHostName | "localhost" |
| -dbServerPort | "50000" |

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 70. Defaulted **manageprofiles** command-line utility parameters

| Parameter | Default values |
|--|---|
| -profilePath | "install_root\profiles\my_WESBDMGR_profile" |
| -hostName | "host_name" |
| -nodeName | "host_nameCellManagernode_number" |
| -cellName | "host_nameCellcell_number" |
|  -winserviceCheck | "true" |
|  -winserviceAccountType | "localsystem" |
|  -winserviceStartupType | "manual" |
|  -winserviceUserName | "Administrator" |
| -dbOutputScriptDir | "install_root\profiles\my_WESBDMGR_profile\dbscripts" |

Custom profile

The following command example creates a custom profile called *my_WESBCUSTOM_profile* on a Windows server.

This example is set to operate with the deployment manager profile created above.

The parameters in Table 71 and Table 72 on page 174 specify the following:

- The DB2 database product will be used for the Common database, which is assumed to already exist. The custom profile creation needs to point to the database used by the deployment manager to which the custom profile will be federated.
- Administrative security will be enabled on the deployment manager to which the custom profile will be federated.
- The custom node will be federated during profile creation for which the deployment manager must be running.

Tip: If the deployment manager is not running, or you want to federate the custom node after profile creation, set the **federateLaterBPM** to true.

See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

WebSphere Enterprise Service Bus example

Specified **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a custom profile.

Table 71. Specified **manageprofiles** command-line utility parameters

| Parameter | Value |
|---------------|---|
| -create | N/A |
| -templatePath | "install_root\profileTemplates\managed.esbserver" (must be fully qualified) |

Table 71. Specified **manageprofiles** command-line utility parameters (continued)

| Parameter | Value |
|--------------------|--|
| -profileName | "my_WESBCUSTOM_profile" |
| -dmgrHost | "remote_host" |
| -dmgrPort | "8879" (To find the -dmgrPort value, go to the <i>dmgr_profile_root</i> \logs directory for the deployment manager associated with this custom profile. In this directory, open the AboutThisProfile.txt file and find the value for the entry "Management SOAP connector port:".) |
| -dmgrAdminPassword | "admin_pwd" |
| -dmgrAdminUserName | "admin_id" |
| -federateLaterWESB | "false" |
| -dbType | "DB2_UNIVERSAL" or "DB2_DATASERVER" |
| -dbJDBCClasspath | "install_root\jdbcdrivers\DB2" |

Remember: If the **federateLaterWESB** parameter is set to true, then the dmgrHost, dmgrPort, dmgrAdminPassword and dmgrAdminUserName do not need to be specified.

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 72. Defaulted **manageprofiles** command-line utility parameters

| Parameter | Default values |
|--------------|---|
| -profilePath | "install_root\profiles\my_WESBCUSTOM_profile" |
| -hostName | "host_name" |
| -nodeName | "host_nameNodenode_number" |

Augmenting profiles

You can augment an existing profile for WebSphere Application Server version 7.0 or WebSphere Application Server Network Deployment version 7.0 to add support for WebSphere Enterprise Service Bus.

Before you begin

- See the list of prerequisites for creating or augmenting profiles in the topic Prerequisites for creating or augmenting profiles.
- Ensure that the profile has the following characteristics:
 - It exists on a system with an installation of WebSphere ESB.
 - It is not federated to a deployment manager. You cannot use the Profile Management Tool or the **manageprofiles** command-line utility to augment federated profiles.
 - It does not have running servers.

About this task

If you have existing WebSphere Application Server or WebSphere Application Server Network Deployment profiles on your system, you might want the operating environments defined by those profiles to have WebSphere ESB functionality.

Restrictions:

- You cannot use the Profile Management Tool to augment profiles on WebSphere ESB installations on 64-bit architectures except on the Linux on zSeries platform. To augment profiles on other 64-bit architectures, you can use the **manageprofiles** command-line utility. For information about using the **manageprofiles** command-line utility, see Augmenting profiles using the **manageprofiles** command-line utility. You can also use the Profile Management Tool on these architectures if you use an WebSphere ESB 32-bit installation.

Use the instructions in this section and its subsections to augment profiles interactively by using the Profile Management Tool graphical user interface (GUI) or, from a command line, by using the **manageprofiles** command-line utility.

Related tasks:

“Migrating the security configuration for a stand-alone environment” on page 358
You can migrate the security configuration and security settings from a previous version of WebSphere ESB to the current version.

Augmenting profiles using the Profile Management Tool:

Use the Profile Management Tool to augment WebSphere Application Server version 7.0 or WebSphere Application Server Network Deployment version 7.0 profiles into WebSphere Enterprise Service Bus profiles.

Before you begin

Ensure that the following prerequisites are satisfied:

- The profile type you will augment to (stand-alone server, deployment manager, or custom) is the same as the type of the profile from which you will augment.
- You have reviewed the list of prerequisites for creating or augmenting profiles at “Prerequisites for creating or augmenting profiles” on page 113.
- You have shut down any servers associated with the profile you plan to augment.
- If you plan to augment a stand-alone server or custom profile, you ensured that it is *not* federated to a deployment manager.

- **Solaris** When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons of the Profile Management Tool. To fix the problem, add the following lines to the *install_root/.Xdefaults* file:

```
Eclipse*spacing:0
```

```
Eclipse*fontList:-misc-fixed-medium-r-normal-*-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

Procedure

1. Start the WebSphere ESB Profile Management Tool.

Use one of the following commands:

- `Linux` `UNIX` `install_root/bin/ProfileManagement/pmt.sh`
- `Windows` `install_root\bin\ProfileManagement\pmt.bat`

See the topic “Starting the Profile Management Tool” on page 118 for other methods of starting this tool.

The Welcome page is displayed.

2. In the Welcome page, click the **Launch Profile Management Tool** button or the **Profile Management Tool** tab.

The **Profiles** tab is displayed.

3. In the **Profiles** tab, highlight the profile you want to augment and click **Augment**.

The **Profiles** tab lists the profiles that exist on your system. For this procedure, it is assumed you are augmenting an existing profile.

Restrictions:

- You cannot augment WebSphere Application Server or WebSphere Application Server Network Deployment version 6.2 profiles into WebSphere ESB version 7.0 profiles.
- You cannot augment cell stand-alone server, management administrative agent, management job manager, or secure proxy profiles.
- If you augment a WebSphere Application Server or WebSphere Application Server Network Deployment profile, it must be from the version of WebSphere Application Server on which WebSphere ESB is installed. The **Augment** button is cannot be selected unless a profile can be augmented.

The Augment Selection page opens in a separate window.

4. In the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.

The Profile Augmentation Options page is displayed.

5. In the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

The **Typical** option augments a profile with default configuration settings.

The **Advanced** option lets you specify your own configuration values for a profile.

Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:

- The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
 - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
 - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.
6. Before continuing to the next page in the Profile Management Tool, proceed to one of the following topics to configure and complete augmentation of your profile.

| Type of profile augmentation you selected | Procedure to complete profile augmentation based on your profile type (stand-alone server, deployment manager, or custom) |
|---|---|
| Typical | <ul style="list-style-type: none"> • “Augmenting stand-alone server profiles using the Typical profile augmentation option” • “Augmenting deployment manager profiles using the Typical augmentation option” on page 182 • “Augmenting custom profiles (managed nodes) using the Typical augmentation option” on page 186 |
| Advanced | <ul style="list-style-type: none"> • “Augmenting stand-alone server profiles using the Advanced profile augmentation option” on page 179 • “Augmenting deployment manager profiles using the Advanced option” on page 184 • “Augmenting custom profiles (managed nodes) using the Advanced augmentation option” on page 188 |

Results

You are ready to configure your profile, which will define an extended operating environment of the type you specified (stand-alone server, deployment manager, or custom).

*Augmenting stand-alone server profiles using the **Typical profile augmentation** option:*

Learn how to use the **Typical profile augmentation** option of the Profile Management Tool to augment and configure WebSphere ESB stand-alone server profiles. Selecting the **Typical** option augments profiles with default configuration settings.

Before you begin

This topic assumes that you are using the Profile Management Tool to augment profiles. As a result, it is assumed that you have started the Profile Management Tool, have chosen to augment a stand-alone server profile, and have selected the **Typical profile augmentation** profile augmentation option.

About this task

In this type of configuration, the Profile Management Tool performs the following tasks:

- Gives option to deploy the administrative console.
- If you are augmenting a profile that has security enabled, lets you enable administrative security on the WebSphere ESB profile you are creating.
- Sets the Common Event Infrastructure and Common database configurations to DB2 Express (if they are not already configured on the profile you are augmenting).

- If you are augmenting a profile that has security enabled, configures Business Space using DB2 Express (if it is not already configured).

Restriction: If you plan to federate the stand-alone server profile to a deployment manager, do not use the **Typical** option to create it. The default values for messaging engine storage and database type provided in a **Typical** profile augmentation are not suitable deployment environment installations. Use the **Advanced** option to augment the profile instead. See “Augmenting stand-alone server profiles using the **Advanced profile augmentation** option” on page 179 for instructions.

As a result of following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175, you are viewing the Administrative Security page, the Profile Summary page, or the Database Security page.

Procedure

1. The page you see displayed in the Profile Management Tool depends on whether security is enabled and databases are configured on that profile.

| State of security and databases on profile | First step |
|--|---|
| <ul style="list-style-type: none"> • Administrative security <i>is</i> enabled on the profile you are augmenting. | The Administrative Security page is displayed. Proceed to step 2. |
| <ul style="list-style-type: none"> • Administrative security is <i>not</i> enabled on the profile you are augmenting. • You do <i>not</i> have databases already configured. | The Profile Summary page is displayed. Proceed to step 3. |
| <ul style="list-style-type: none"> • Administrative security is <i>not</i> enabled on the profile you are augmenting. • You <i>do</i> have databases already configured. | A password page asks for the database user name and password used to configure the databases. Enter the information and click Next . The Profile Summary page is displayed. Proceed to step 3. |

2. Enable administrative security.

If you see this page, the profile you are augmenting has security enabled. You must re-enter the administrative user ID and password for that profile.

If the profile you are augmenting has the WebSphere Application Server sample application deployed, it requires an account under which to run. Supply the password for the account. You cannot change the user name of the account.

The Profile Summary page is displayed.

3. In the Profile Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

When the profile augmentation is complete, the Profile Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

Attention: If errors are detected during profile augmentation, other messages might appear in place of the success message, for example:

- **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
- **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

4. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console to start the server.

Results

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere Enterprise Service Bus profile.

The node within the profile has a server named server1 on Linux, UNIX, and Windows platforms. The server number is incremented if there is more than one product installation.

What to do next

Check the server operation by selecting **Start the server** from the First steps console. An output window opens. If you see a message like the following one, your server is operating properly:

```
ADMU3000I: Server server1 open for e-business; process id is 3348
```

You can also check server operation by running the Installation Verification Test (IVT) from the First steps console or running the `wbi_ivt` command-line utility. This test is to verify that your deployment manager or stand-alone server installation is operating properly. For a stand-alone server profile, it also runs a System Health check and generates a report.

*Augmenting stand-alone server profiles using the **Advanced profile augmentation** option:*

Learn how to use the **Advanced** option of the Profile Management Tool to augment and configure WebSphere ESB stand-alone server profiles. Selecting the **Advanced** option augments profiles with customized configuration settings.

Before you begin

This topic assumes that you are using the Profile Management Tool to augment profile. As a result, it is assumed that you have started the Profile Management Tool, have chosen to augment a stand-alone server profile, and have selected the **Advanced** profile augmentation option.

About this task

By selecting the **Advanced** option, you can perform the following tasks:

- Configure the Common Event Infrastructure.
- Configure the Common database.
- If you are augmenting a profile that has security enabled, enable administrative security on the WebSphere ESB profile you are creating.
- If you are augmenting a profile that has security enabled, configure Business Space using DB2 Express.
- Configure the databases using a database design file.

Important: The procedure in this topic outlines all the pages available in the Profile Management Tool to configure an Advanced stand-alone server profile. If

particular components, such as the Common database or Business Space, are already configured on the profile you are augmenting, configuration pages for those components will not appear.

One of the following pages is displayed: the Administrative Security page or the Business Space Configuration page.

Procedure

1. The page you see displayed in the Profile Management Tool depends on whether the profile you are augmenting has security enabled .

| Profile type you are augmenting to and security status of existing profile you are augmenting | First step |
|---|--|
| <ul style="list-style-type: none"> • WebSphere ESB profile • Security <i>is</i> enabled on the profile you are augmenting. | The Administrative Security page is displayed. Proceed to step 2. |
| <ul style="list-style-type: none"> • WebSphere Enterprise Service Bus profile • Security <i>is not</i> enabled on the profile that you are augmenting | The Business Space Configuration page is displayed. Proceed to step 3. |

2. Enable administrative security.

If you see this page, the profile you are augmenting has security enabled. You must re-enter the administrative user ID and password for that profile.

If the profile you are augmenting has the WebSphere Application Server sample application deployed, it requires an account under which to run. Supply the password for the account. You cannot change the user name of the account.

The Business Space Configuration page is displayed. Proceed to step 3.

3. On the Business Space Configuration page, select the **Configure Business Space** check box to set up Business Space powered by WebSphere, an integrated user experience for application users across the IBM WebSphere business process management portfolio. Then click **Next**. Configuring Business Space sets up an integrated GUI for the business users of your application for this profile.

Important: Business Space is supported with the following database products: DB2 Express, DB2 Universal, DB2 for i5/OS (DB2 for IBM i), DB2 for z/OS, Oracle, and Microsoft SQL Server 2005 and 2008.

If the database you use for WebSphere ESB does not match the supported databases for Business Space, a DB2 Express database is selected for the Business Space configuration. You cannot federate this profile into a deployment environment later, because DB2 Express is not supported for deployment environments.

The next step depends on whether databases are already defined on your system.

Table 73. Next step based on whether databases are configured

| Condition of databases | Next step |
|--|---|
| <ul style="list-style-type: none"> • Databases are <i>not</i> already defined on your system. | The Database Design page is displayed. Proceed to step 4 on page 181. |

Table 73. Next step based on whether databases are configured (continued)

| Condition of databases | Next step |
|--|---|
| <ul style="list-style-type: none"> Databases <i>are</i> already defined on your system. | A password page asks for the database user name and password used to configure the databases. Enter the information and click Next . The Profile Summary page is displayed. Proceed to step 6. |

4. Optional: Configure the databases using a design file. This option is available for both stand-alone server and deployment manager profiles created using the **Advanced** option.
 - a. Select **Use a database design file for database configuration**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. Click **Next**.

If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration. For more information on using a design file for database configuration, see “Creating database design files by using the database design tool” on page 97.

5. In the Database Configuration page, configure the Common database used by selected WebSphere ESB components.

Refer to Creating a stand-alone profile by using the Profile Management Tool for details about the fields on the Database Configuration page and the Database Configuration (Part 2) page, then return to this step when you have completed.

The Profile Summary page is displayed.

6. In the Profile Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

When the profile augmentation is complete, the Profile Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

Attention: If errors are detected during profile augmentation, other messages might appear in place of the success message, for example:

- **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
- **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

7. Complete the stand-alone server profile configuration by performing one of the following tasks, depending on whether you must manually configure the Common Event Infrastructure and Common databases.
 - If you completed configuration of the Common Event Infrastructure and Common databases using the Profile Management Tool, ensure **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console to start the server.

- If you chose to postpone actual database configuration by producing scripts to be run manually, perform the following steps:
 - a. Clear the check box beside **Launch the First steps console** and click **Finish** to close the Profile Management Tool. Also, close the Profiles page, which is open in a separate window.
 - b. Use your site's standard database definition tools and procedures to edit and run the scripts the Profile Management Tool generated to create, or create and configure, the event, eventcat, and WPRCSDB databases (or their equivalents if they have different names on your system). You identified the location for these scripts in an earlier step, when configuring database details. Also see the topics that describe manually creating new databases or new tables in existing databases:
 - For the Common Event Infrastructure database: Configuring the event database and its subtopics.
 - For the Common database: “Creating the Common database and tables after profile creation or augmentation” on page 95.

Results

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere Enterprise Service Bus profile.

If you used the default server name, the node within the profile has a server named `server1` for Linux, UNIX, and Windows platforms and the number is incremented if there is more than one product installation.

What to do next

Start the First steps console associated with the profile, as described in Starting the First steps console.

Check server operation by selecting **Start the server** from the First steps console. An output window opens. If you see a message like the following message, your server is operating properly:

```
ADMU3000I: Server server1 open for e-business; process id is 3348
```

You can also check server operation by running the Installation Verification Test (IVT) from the First steps console or running the **wbi_ivt** command-line utility. This test is to verify that your deployment manager or stand-alone server installation is operating properly. For a stand-alone server profile, it also runs a System Health check and generates a report.

*Augmenting deployment manager profiles using the **Typical** augmentation option:*

Learn how to use the **Typical** option of the Profile Management Tool to augment and configure WebSphere ESB deployment manager profiles. Selecting the **Typical** option augments profiles with default configuration settings.

Before you begin

This topic assumes that you are using the Profile Management Tool to augment profiles and are following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175. As a result, it is assumed that you have started the Profile Management Tool, have chosen to augment a deployment manager

profile, and have selected the **Typical** profile augmentation option.

About this task

In this type of configuration, the Profile Management Tool performs the following tasks:

- If you are augmenting a profile that has security enabled, lets you enable administrative security on the WebSphere ESB profile you are creating.
- Sets the Common database configuration to DB2 Express (if it is not already configured on the profile you are augmenting).

As a result of following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175, you are viewing the Administrative Security page, the Profile Summary page, or the Database Security page.

Procedure

1. The page you see in the Profile Management Tool depends on whether administrative security is enabled on the profile.

| State of security and databases on profile | First step |
|---|---|
| <ul style="list-style-type: none">• Administrative security <i>is</i> enabled on the profile you are augmenting. | The Administrative Security page is displayed. Proceed to step 2. |
| <ul style="list-style-type: none">• Administrative security is <i>not</i> enabled on the profile you are augmenting.• You do <i>not</i> have databases already configured. | The Profile Summary page is displayed. Proceed to step 3. |
| <ul style="list-style-type: none">• Administrative security is <i>not</i> enabled on the profile you are augmenting.• You <i>do</i> have databases already configured. | A password page asks for the database user name and password used to configure the databases. Enter the information and click Next . The Profile Summary page is displayed. Proceed to step 3. |

2. Enable administrative security.

If you see this page, the profile you are augmenting has security enabled. You must re-enter the administrative user ID and password for that profile.

The Profile Summary page is displayed.

3. In the Profile Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

When the profile augmentation is complete, the Profile Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

Attention: If errors are detected during profile augmentation, other messages might appear in place of the success message, for example:

- **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
- **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

4. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console start the server.

Results

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere Enterprise Service Bus profile.

The node defined by the profile has a deployment manager named Dmgr.

What to do next

Check the server operation by selecting **Start the deployment manager** from the First steps console. An output window opens. If you see a message like the following one, your deployment manager is operating properly:

```
ADMU3000I: Server dmgr open for e-business; process id is 3072
```

In a deployment environment, you must create and configure other databases, create custom profiles, and federate them to your deployment manager, create servers, create clusters if you want workload management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

*Augmenting deployment manager profiles using the **Advanced** option:*

Learn how to use the **Advanced** option of the Profile Management Tool to augment and configure WebSphere ESB deployment manager profiles. Selecting the **Advanced** option augments profiles with customized configuration settings.

Before you begin

This topic assumes that you are using the Profile Management Tool to augment profiles and are following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175. As a result, it is assumed that you have started the Profile Management Tool, have chosen to augment a deployment manager profile, and have selected the **Advanced** profile augmentation option.

About this task

By selecting the **Advanced** option, you can perform the following tasks:

- Configure the Common database.
- Configure the database using a database design file.
- If you are augmenting a profile that has security enabled, enable administrative security on the WebSphere ESB profile you are creating.

Important: The procedure in this topic outlines all the pages available in the Profile Management Tool to configure an Advanced deployment manager profile. If a particular component, such as the Common database, is already configured on the profile you are augmenting, the configuration page for that component will not appear.

As a result of following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175, one of the following pages is displayed: the

Administrative Security page, the Database Design page, or the Database Security page.

Procedure

1. The page you see in the Profile Management Tool depends on whether administrative security is enabled on the profile and on whether the Common database is already configured.

| State of security and database on profile | First step |
|---|--|
| <ul style="list-style-type: none">• Administrative security <i>is</i> enabled on the profile you are augmenting. | The Administrative Security page is displayed. Proceed to step 2. |
| <ul style="list-style-type: none">• Administrative security is <i>not</i> enabled on the profile you are augmenting.• You do <i>not</i> have the Common database already configured. | The Database Design page is displayed. Proceed to step 3. |
| <ul style="list-style-type: none">• Administrative security is <i>not</i> enabled on the profile you are augmenting.• You <i>do</i> have the Common database already configured. | A password page asks for the database user name and password used to configure the database. Enter the information and click Next . The Profile Summary page is displayed. Proceed to step 5. |

2. Enable administrative security.

If you see the Administrative Security page, the profile you are augmenting has security enabled. You must reenter the administrative user ID and password for that profile.

The Database Configuration page is displayed.

3. Optional: Configure the database using a design file. This option is available for both Advanced stand-alone server and Advanced deployment manager profiles.
 - a. Select **Use a database design file for database configuration**.
 - b. Click **Browse**.
 - c. Specify the fully qualified path name for the design file.
 - d. Click **Next**.

If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration. For more information on using a design file for database configuration, see “Creating database design files by using the database design tool” on page 97.

4. In the Database Configuration page, configure the Common database used by the selected product components.

See the topic Creating a stand-alone profile by using the Profile Management Tool for details about the fields on the Database Configuration and Database Configuration (Part 2) pages, then return to this step when you have completed those pages. The Profile Summary page is displayed.

5. In the Profile Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

When the profile augmentation is complete, the Profile Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

Attention: If errors are detected during profile augmentation, other messages might appear in place of the success message, for example:

- **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
- **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

6. Complete the profile configuration by doing one of the following tasks, depending on whether you must manually configure the Common database.
 - If you completed configuration of the Common database using the Profile Management Tool, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console to start the deployment manager.
 - If you decided to postpone actual database configuration by producing scripts to be run manually, perform the following steps:
 - a. Clear the check box beside **Launch the First steps console** and click **Finish** to close the Profile Management Tool. Also, close the Profiles page, which is open in a separate window.
 - b. Use your site's standard database definition tools and procedures to edit and run the scripts the Profile Management Tool generated to create or create and configure the WPRCSDB database (or its equivalent if it has a different name on your system). You identified the location for this script when you configured the database details in an earlier step.

Results

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere ESB profile.

What to do next

Start the First steps console associated with the profile, as instructed in Starting the First steps console.

Check server operation by selecting **Start the deployment manager** from the First steps console. An output window opens. If you see a message like the following one, your deployment manager is operating properly:

```
ADMU3000I: Server dmgr open for e-business; process id is 3072
```

In a deployment environment, you must create and configure other databases, create custom profiles, and federate them to your deployment manager, create servers, create clusters if you want workload management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

*Augmenting custom profiles (managed nodes) using the **Typical** augmentation option:*

Learn how to use the **Typical** option of the Profile Management Tool to augment and configure WebSphere ESB custom profiles. Selecting the **Typical** option augments profiles with default configuration settings.

Before you begin

This topic assumes that you are using the Profile Management Tool to augment profiles and are following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175. As a result, it is assumed that you have started the Profile Management Tool, have chosen to augment a custom profile, and have selected the **Typical** profile augmentation option.

About this task

In this type of configuration, you can choose to federate the node to an existing deployment manager during the augmentation process, or federate it later using the **addNode** command. If you decide to federate the profile during the augmentation process, the tool sets the Common database configuration to the same database as the deployment manager. If you decide not to federate, the database configuration is left unconfigured.

As a result of following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175, the Federation page is displayed.

Procedure

1. In the Federation page, choose to federate the node into the deployment manager now as part of the profile augmentation, or at a later time and apart from profile augmentation.

- If you choose to federate the node as part of the profile augmentation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if administrative security is enabled on the deployment manager. Leave the **Federate this node later** check box deselected. Then click **Next**.

The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).

Attention: Federate the custom node during profile augmentation only if all the following conditions are true:

- You do not plan to use this custom node as a migration target.
- No other node is being federated. (Node federation must be serialized.)
- The deployment manager is running.
- The deployment manager is an WebSphere ESB deployment manager.
- The deployment manager is at a release level the same or higher than the release level of the custom profile you are augmenting.
- The deployment manager has a JMX administrative port enabled. The default protocol is SOAP. (Click **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

- If you choose to federate the node at a later time and apart from profile augmentation, select the **Federate this node later** check box and click **Next**. See “Federating custom nodes to a deployment manager” on page 146 for more information about how to federate a node by using the **addNode** command. Read more about this command in the Using wsadmin scripting

to run the addNode command topic in the WebSphere Application Server Network Deployment information center.

The Profile Summary page is displayed.

2. In the Profile Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

When the profile augmentation is complete, the Profile Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

Attention: If errors are detected during profile augmentation, other messages might appear in place of the success message, for example:

- **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
- **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

3. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.

Results

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere Enterprise Service Bus profile.

What to do next

If you did not federate the profile during profile augmentation, federate it now. The node within the profile is empty until you federate the node and use the deployment manager to customize the node.

Augmenting custom profiles (managed nodes) using the Advanced augmentation option:

Learn how to use the **Advanced** option of the Profile Management Tool to augment and configure WebSphere ESB custom profiles. Selecting the **Advanced** option augments profiles with customized configuration settings.

Before you begin

This topic assumes that you are using the Profile Management Tool to augment profiles and are following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175. As a result, it is assumed that you have started the Profile Management Tool, have chosen to augment a custom profile, and have selected the **Advanced** profile augmentation option.

About this task

While augmenting custom profiles, you can choose to federate the node to an existing deployment manager during the augmentation process, or federate it later using the **addNode** command.

As a result of following the procedure in “Augmenting profiles using the Profile Management Tool” on page 175, you are viewing the Federation page.

Procedure

1. In the Federation page, choose to federate the node into the deployment manager now as part of the profile augmentation, or at a later time and apart from profile augmentation.

- If you choose to federate the node as part of the profile augmentation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password (if administrative security is enabled on the deployment manager). Leave the **Federate this node later** check box deselected. Then click **Next**.

The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).

Important:

Do *not* federate the custom node during profile augmentation if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has not yet been augmented into an WebSphere ESB deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the custom profile you are creating or augmenting.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is reconfigured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

- If you choose to federate the node at a later time and apart from profile augmentation, select the **Federate this node later** check box and click **Next**. See “Federating custom nodes to a deployment manager” on page 146 for more information about how to federate a node by using the **addNode** command. Read more about this command in the Using wsadmin scripting to run the addNode command topic in the WebSphere Application Server Network Deployment information center.

The Database Configuration page is displayed.

2. Optional: Configure the databases using a design file. This option is available for both stand-alone server and deployment manager profiles created using the **Advanced** option.
 - a. Select **Use a database design file for database configuration**.
 - b. Click **Browse**.

- c. Specify the fully qualified path name for the design file.
- d. Click **Next**.

If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration. For more information on using a design file for database configuration, see “Creating database design files by using the database design tool” on page 97.

3. In the Database Configuration page, configure the Common database used by selected WebSphere ESB components.

Refer to Creating custom profiles (managed nodes), by using the Profile Management Tool for details about the fields on the Database Configuration page and the Database Configuration (Part 2) page, then return to this step when you have completed.

The Profile Summary page is displayed.

4. In the Profile Summary page, click **Augment** to augment the profile, or **Back** to change the characteristics of the profile.

When the profile augmentation is complete, the Profile Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

Attention: If errors are detected during profile augmentation, other messages might appear in place of the success message, for example:

- **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
- **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

5. In the Profile Complete page, ensure **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access product documentation.

Results

Augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere Enterprise Service Bus profile.

What to do next

The node within the profile is empty until you federate it and use the administrative console to customize it.

In a deployment environment, you must create and configure databases, create other custom profiles and federate them to your deployment manager, create servers, create clusters if you want workload management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

Augmenting profiles using the manageprofiles command-line utility:

Augmentation is the ability to change an existing profile with an augmentation template. You can augment a profile by using the **manageprofiles** command-line utility.

Before you begin

Before using this procedure, ensure that you have done the following tasks:

- You have reviewed the list of prerequisites for creating or augmenting a profile at Prerequisites for creating or augmenting profiles.
- You have shut down any servers associated with the profile that you plan to augment.
- If you plan to augment a stand-alone server or custom profile, you have determined if it has already been federated to a deployment manager:
 - If the profile you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.
 - If the profile you want to augment has not already been federated to a deployment manager, when you do federate it later by using the **addNode** command, the following must be true of the deployment manager with which it is federated in order for the augmentation to complete successfully:
 - It must be running.
 - It must be an WebSphere Enterprise Service Bus deployment manager profile.
 - It must be at a release level the same or higher than that of the profile you are augmenting.
 - WebSphere Enterprise Service Bus profiles can use a WebSphere Enterprise Service Bus deployment manager or IBM Business Process Manager deployment manager.
 - It must have a JMX administrative port enabled. The default protocol is SOAP.
- You have reviewed the **manageprofiles** parameters and default values.
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Security role required for this task: See Granting write permission of files and directories to nonroot users for profile creation .

To use the **manageprofiles** command-line utility to augment a profile, perform the following steps.

Procedure

1. Determine the template that the existing profile was created with (deployment manager, stand-alone, or managed). You can determine the template that was used for creating the profile by viewing the profile registry in *install_root/properties/profileRegistry.xml*. Do not modify this file, use it only to view the templates.
2. Find the appropriate template to augment to. You can augment an existing WebSphere Application Server or WebSphere Application Server Network Deployment profile into an IBM Business Process Manager or WebSphere ESB profile. The following profile templates are available:

- `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile, which defines a stand-alone server.
- `dmgr.esbserver`: for a WebSphere Enterprise Service Bus deployment manager profile, which defines a deployment manager.
- `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile, which, when federated to a deployment manager, defines a managed node.

Use the `augment` parameter to make changes to an existing profile with an augmentation template. The `augment` parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the `install_dir/profileTemplates` directory.

3. Run the file from the command line. Do not supply a **-profilePath** parameter. Here are some simple examples of augmenting Process Server profiles.

- **Linux** **UNIX** `manageprofiles.sh -augment -templatePath install_root/profileTemplates/default.esbserver -profileName MyProfileName`
- **Windows** `manageprofiles.bat -augment -templatePath install_root\profileTemplates\default.esbserver -profileName MyProfileName`

If you have created a response file, use the **-response** parameter: `-response myResponseFile`

The following example shows a response file for an `augment` operation:

```
augment
profileName=testResponseFileAugment
templatePath=install_root/profileTemplates/default.esbserver
```

```
nodeName=myNodeName
cellName=myCellName
hostName=myHostName
omitAction=myOptionalAction1, myOptionalAction2
```

The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

What to do next

You can see that your profile augmentation completed successfully if you receive a `INSTCONFSUCCESS: Profile augmentation succeeded.` message, and you can check the following log file:

- **Linux** **UNIX** `install_root/logs/manageprofiles/profile_name_augment.log`
- **Windows** `install_root\logs\manageprofiles\profile_name_augment.log`

Run the Installation Verification Test (IVT) tool to verify that the profile was augmented successfully. To do this, run the following command:

- **Linux** **UNIX** On Linux and UNIX platforms: `profile_root/bin/wbi_ivt.sh`
- **Windows** On Windows platforms: `profile_root\bin\wbi_ivt.bat`

Augmenting stand-alone profiles using the `manageprofiles` command-line utility:

Augmentation is the ability to change an existing profile with an augmentation template. You can augment existing WebSphere Application Server or WebSphere Application Server Network Deployment profiles into WebSphere ESB profiles. You can augment a profile from the command line using the **`manageprofiles`** command-line utility.

Before you begin

- You have reviewed the list of prerequisites for creating or augmenting a profile at “Prerequisites for creating or augmenting profiles” on page 113.
- You have shut down any servers associated with the profile that you plan to augment.
- If you plan to augment a stand-alone server or custom profile, you have determined if it has already been federated to a deployment manager:
 - If the profile you want to augment has already been federated to a deployment manager, you cannot augment it using the **`manageprofiles`** command-line utility.
 - If the profile you want to augment has not already been federated to a deployment manager, when you do federate it via the **`addNode`** command later, the following must be true of the deployment manager with which it is federated in order for the augmentation to complete successfully:
 - It must be running.
 - It must be at a release level the same or higher than that of the profile you are augmenting. WebSphere Enterprise Service Bus profiles can use a WebSphere Enterprise Service Bus or IBM Business Process Manager Process Server deployment manager.
 - It must have a JMX administrative port enabled. The default protocol is SOAP.
 - It must have already been augmented into an IBM Business Process Manager or WebSphere Enterprise Service Bus profile, depending on the product you have installed.
- You have reviewed example profile augmentation commands in this section.
- You have verified that you are not already running the **`manageprofiles`** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Security role required for this task: See “Granting write permission of files and directories to nonroot users for profile creation” on page 113.

Determine the template that the existing profile was created with (deployment manager, stand-alone, or managed). You can determine the template that was used for creating the profile by viewing the profile registry in `install_root/properties/profileRegistry.xml`. Do not modify this file, use it only to view the templates. For this procedure it is assumed that you are augmenting a stand-alone profile.

About this task

This task describes how to use **manageprofiles** to augment stand-alone profiles.

To use the **manageprofiles** command-line utility to augment a stand-alone profile, perform the following steps.

Procedure

1. Find the appropriate template to augment to. You can augment an existing WebSphere Application Server or WebSphere Application Server Network Deployment profile into an IBM Business Process Manager or WebSphere ESB profile. The following profile templates are available:
 - **default.esbserver**: for a WebSphere Enterprise Service Bus stand-alone server profile, which defines a stand-alone server.

Use the **augment** parameter to make changes to an existing profile with an augmentation template. The **augment** parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

Note: Do not manually modify the files that are located in the *install_dir/profileTemplates* directory.

2. Run the file from the command line. Do not supply a **-profilePath** parameter. Here are some simple examples.

- **Linux** **UNIX** `manageprofiles.sh -augment -templatePath install_root/profileTemplates/default.esbserver -profileName MyProfileName`
- **Windows** `manageprofiles.bat -augment -templatePath install_root\profileTemplates\default.esbserver -profileName MyProfileName`

If you have created a response file, use the **-response** parameter: **-response myResponseFile**

The following example shows a response file for an **augment** operation:

```
augment
profileName=testResponseFileAugment
templatePath=install_root/profileTemplates/default.esbserver

nodeName=myNodeName
cellName=myCellName
hostName=myHostName
omitAction=myOptionalAction1, myOptionalAction2
```

The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

What to do next

You can see that your profile augmentation completed successfully if you receive a **INSTCONFSUCCESS: Profile augmentation succeeded.** message, and you can check the following log file:

- **Linux** **UNIX** `install_root/logs/manageprofiles/
profile_name_augment.log`
- **Windows** `install_root\logs\manageprofiles\profile_name_augment.log`

Run the Installation Verification Test (IVT) tool to verify that the profile was augmented successfully. To do this, run the following command:

- **Linux** **UNIX** **On Linux and UNIX platforms:** `profile_root/bin/
wbi_ivt.sh`
- **Windows** **On Windows platforms:** `profile_root\bin\wbi_ivt.bat`

Database configuration fields for Profile Management Tool configuration:

To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Refer to one of the following tables to determine the required database parameters for your specific database type.

Table 74. Database configuration parameters for Profile Management Tool configuration

| Database type |
|--|
| DB2 Universal Database configuration fields |
| DB2 Universal Database for z/OS configuration fields |
| Microsoft SQL Server database configuration fields |
| Oracle database configuration fields |
| Oracle database configuration fields |

DB2 Universal Database configuration fields

The following table lists the fields you must complete on the Database Configuration - Part 2 page when you select DB2 Universal Database as your database product.

Table 75. Required database configuration fields for DB2 Database

| Field | Action needed |
|---|--|
| JDBC driver | Select from the following options: <ul style="list-style-type: none"> • DB2 Universal • DB2 DataServer |
| User name for database authentication | Enter the user name to authenticate with the database. |
| Password for database authentication | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Directory location of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in one of the following directories: <ul style="list-style-type: none"> • If you selected the DB2 Express feature during the installation: <code>\${WAS_INSTALL_ROOT}/db2/java</code> • If you did not select the DB2 Express feature during the installation: <code>\${WAS_INSTALL_ROOT}/jdbcd drivers/DB2</code> |

Table 75. Required database configuration fields for DB2 Database (continued)

| Field | Action needed |
|--|---|
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 50000 or enter the correct server port number. |

In a stand-alone configuration, when you configure the Process Server database, the Process Server messaging engine tables are created in the Process Server database.

DB2 Universal Database for z/OS configuration fields

The following table lists the fields you must complete on the Database Configuration - Part 2 page when you select DB2 Universal Database for z/OS as your database product.

Table 76. Required database configuration fields for DB2 Universal Database for z/OS

| Field | Action needed |
|--|---|
| User name for database authentication | Enter the user name to authenticate with the database. |
| Password for database authentication | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Directory location of JDBC driver classpath files | The JDBC driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}/jdbcdrivers/DB2 |
| Database server host name (for example IP address) | Enter the database server host name. |
| Server port | Accept the default value of 446 or enter the correct server port number. |
| Process Server Schema name | Enter the database schema name for Process Server. |
| Performance Server Schema name | Enter the database schema name for Performance Server. |
| Connection location | Enter the connection location. |
| Storage group name | Enter the storage group name. |

You cannot create a new database using DB2 Universal Database for z/OS. .

Microsoft SQL Server database configuration fields

The following table lists the fields you must complete on the Database Configuration - Part 2 page when you select Microsoft SQL Server as your database product.

Table 77. Required database configuration fields for Microsoft SQL Server

| Field | Action needed |
|---|---------------|
| Select the Apply Windows Authentication option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Common database, Process Server database, and Performance Data Warehouse database fields are made inactive. | |

Table 77. Required database configuration fields for Microsoft SQL Server (continued)

| Field | Action needed |
|---|--|
| Common database | <p>For the Common database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Common database user name. • Password Enter a password to authenticate with the Common database. • Confirm password Confirm the password to authenticate with the Common database. |
| Process Server database | <p>For the Process Server database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Process Server database user name. • Password Enter a password to authenticate with the Process Server database. • Confirm password Confirm the password to authenticate with the Process Server database. |
| Performance Data Warehouse database | <p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <ul style="list-style-type: none"> • User name Enter the Performance Data Warehouse database user name. • Password Enter a password to authenticate with the Performance Data Warehouse database. • Confirm password Confirm the password to authenticate with the Performance Data Warehouse database. |
| Directory location of JDBC driver classpath files | The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1433 or enter the correct server port number. |

The following table shows the required fields for Microsoft SQL Server configuration on the Database Configuration - Part 3 page. If you are creating a deployment manager and selected the **Apply Windows Authentication** option, this page is not displayed.

Table 78. Required database configuration fields for Microsoft SQL Server

| Field | Action needed |
|---|---|
| Common database schema name | Enter the schema name for the Common database. This field is not displayed if you are creating a stand-alone profile and selected the Apply Windows Authentication option. |
| CEI bus messaging engine schema name | Enter the schema name for the Common Event Infrastructure bus messaging engine. |
| SCA applications bus messaging engine schema name | Enter the schema name for the Service Component Architecture application bus messaging engine. |
| SCA system bus messaging engine schema name | Enter the schema name for the Service Component Architecture system bus messaging engine. |

Oracle database configuration fields

The following table lists the fields you must complete when you select Oracle as your database product.

Table 79. Required database configuration fields for Oracle

| Field | Action needed |
|--|---|
| Process Server database | <p>For the Process Server database, enter values for the following parameters:</p> <p>User name Enter the Process Server database user name.</p> <p>Password Enter a password to authenticate with the Process Server database.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.</p> |
| Performance Data Warehouse database | <p>For the Performance Data Warehouse database, enter values for the following parameters:</p> <p>User name Enter the Performance Data Warehouse database user name.</p> <p>Password Enter a password to authenticate with the Performance Data Warehouse database.</p> <p>Confirm password Confirm the password by reentering it.</p> <p>Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.</p> |
| Database server host name (for example IP address) | Accept the default value of localhost or enter the correct database server host name. |
| Server port | Accept the default value of 1521 or enter the correct server port number. |

Table 79. Required database configuration fields for Oracle (continued)

| Field | Action needed |
|---|--|
| Directory location of JDBC driver classpath files | The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: \${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle |

The following table shows the required fields for Oracle configuration on the Database Configuration - Part 3 page.

Table 80. Required database configuration fields for Oracle

| Field | Action needed |
|--------------------------------------|--|
| SCA application bus messaging engine | For the SCA application bus messaging engine, enter values for the following parameters: User name Enter the SCA application bus messaging engine user name. Password Enter a password to authenticate with the SCA application bus messaging engine. Confirm password Confirm the password by reentering it. |
| SCA system bus messaging engine | For the SCA system bus messaging engine, enter values for the following parameters: User name Enter the SCA system bus messaging engine user name. Password Enter a password to authenticate with the SCA system bus messaging engine. Confirm password Confirm the password by reentering it. |
| Process Server bus messaging engine | For the Process Server bus messaging engine, enter values for the following parameters: User name Enter the Process Server bus messaging engine user name. Password Enter a password to authenticate with the Process Server bus messaging engine. Confirm password Confirm the password by reentering it. |

Table 80. Required database configuration fields for Oracle (continued)

| Field | Action needed |
|--|---|
| Performance Data Warehouse bus messaging engine | <p>For the Performance Data Warehouse bus messaging engine, enter values for the following parameters:</p> <p>User name Enter the Performance Data Warehouse bus messaging engine user name.</p> <p>Password Enter a password to authenticate with the Performance Data Warehouse bus messaging engine.</p> <p>Confirm password Confirm the password by reentering it.</p> |

DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox) configuration fields

The following table lists the fields you must complete when you select DB2 for i5/OS (Toolbox) or DB2 for IBM i (Toolbox) as your database product.

Table 81. Required database configuration fields for DB2 for i5/OS (Toolbox) and DB2 for IBM i (Toolbox)

| Field | Action needed |
|--|--|
| User name to authenticate with the database | Enter the user name to authenticate with the database. |
| Password for database authentication | Enter a password to authenticate with the database. |
| Confirm password | Confirm the password. |
| Location (directory) of JDBC driver classpath files | <p>Perform one of the following actions:</p> <ul style="list-style-type: none"> Accept the default value of /QIBM/ProdData/HTTP/Public/jt400/lib Browse to the location on your system that contains the following file: jt400.jar <p>An error message is displayed if the file cannot be found at the specified location.</p> |
| Database server host name (for example IP address) | Enter the database server host name. |
| Database collection name | Accept the default value of WPRCSDB or enter the correct schema name. To prevent naming conflicts within the specified database, specify a schema name whose first three characters are unique from the names of other schemas residing in the database. |

Configuring databases

Before starting a profile, you must have configured the databases that are to be used with the profile.

Before you begin

You must have planned your database requirements, including a list of all databases and schema names. For more information, see Planning your database configuration

Configuring a Microsoft SQL Server database

You can create a stand-alone profile for use with Microsoft SQL Server.

Prerequisites

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

Database restrictions

- The databases that are created for the components must be case-sensitive.

Database privileges and security considerations

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

Table 45 on page 91 shows the database privileges that are required to access the data store.

Table 82.

| Database management system | Minimum privilege required to use the data store tables | Additional privilege required to create the data store tables |
|----------------------------|--|---|
| Microsoft SQL Server | Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements. | The user ID requires the CREATE TABLE statement privilege. |

Related tasks:



Configuring an existing database during a typical installation

Use the information in this topic to determine the correct database values for configuring your existing database during a typical installation.

“Configuring XA transactions” on page 92

You must configure XA transactions after the database is installed and before you start the server. The Microsoft SQL Server JDBC Driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the `SQLServerXADataSource` class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Configuring XA transactions:

You must configure XA transactions after the database is installed and before you start the server. The Microsoft SQL Server JDBC Driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the `SQLServerXADataSource` class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

About this task

Failure to configure the XA transactions can result in the following error during server start up: `javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA`

control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'".

Procedure

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

- a. Select **Control Panel > Administrative Tools > Component Services**.
- b. Select **Component Services > Computers** and right-click **My Computer**, and select **Properties**.
- c. Click the **MSDTC** tab, and then click **Security Configuration**.
- d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.
- e. Click **OK** again to close the **Properties** dialog box, and then close **Component Services**.
- f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

- a. Select **Control Panel > Administrative Tools > Component Services**.
 - b. Select **Component Services > Computers > My Computer > Distributed Transaction Coordinator**.
 - c. Right-click **Local DTC** and then select **Properties**.
 - d. Click the **Security** tab on the **Local DTC Properties** dialog box.
 - e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.
 - f. Click **OK** again to close the **Properties** dialog box, and then close **Component Services**.
 - g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.
2. Configure the JDBC Distributed Transaction Components:
 - a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from Microsoft Site using URL from Resources section.
 - b. Unzip archive to any folder.
 - c. Copy the `sqljdbc_xa.dll` file from the JDBC unarchived directory to the `Binn` directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the `sqljdbc_xa.dll` file in the `x86` folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the `sqljdbc_xa.dll` file in the `x64` folder.
 - d. Execute the `xa_install.sql` database script on SQL Server . This script installs the extended stored procedures that are called by `sqljdbc_xa.dll`. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.
 - e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the `SqlJDBCXAUser` role in master database (e.g. for lombardi user add master database in User mappings and check `SqlJDBCXAUser` role).

Related concepts:

“Configuring a Microsoft SQL Server database” on page 91
You can create a stand-alone profile for use with Microsoft SQL Server.

Creating network deployment environments for use with Microsoft SQL Server:

This topic describes how to create a network deployment environment for use with Microsoft SQL Server.

Before you begin

Before creating a profile, complete the following prerequisites:

- Install Microsoft SQL Server on the server that hosts the database.
- If you are going to use the Common Event Infrastructure, you must create the CEI database manually. See *Configuring a Common Event Infrastructure (CEI) database*.

WebSphere ESB packages JDBC drivers for SQL Server. For information about the JDBC drivers (including version and level information), see the page.


Note: You are responsible for providing JDBC driver levels outside of what is packaged with WebSphere ESB.

About this task

You can configure the CommonDB when you create the Deployment Manager profile (WebSphere ESB Advanced only); however, the remaining components must be configured using the Deployment Environment panels in the administrative console. The components to be configured are:

- Common Event Infrastructure
- Business Space
- Messaging Engines

Procedure

1. Create the Deployment Manager profile For more information, see Creating the deployment manager profile.
2. Start the deployment manager using one of the following methods:
 -  From the **Start** menu, select **IBM > Enterprise Service Bus 7.5 > Profiles > *profile_name* > Start the deployment manager**.
 - In the First Steps console, click **Start the deployment manager**.
 - Use the **startManager** command.
3. Create at least one node (managed profile) for use in the deployment environment. For more information, see “Creating deployment manager and custom profiles using manageprofiles” on page 152.
4. Create the deployment environment:
 - a. In the administrative console, select **Servers > Deployment Environment**.
 - b. Click **New**.
 - c. Provide the information for each step until the step to configure the database.
 - d. On the Database page, update the default values for the components that your environment is using.

Make sure you enter the correct values for the user name and schema name for components below. Deployment Environment configuration does not create any schemas and users as part of configuration. These should exist before the generation of the deployment environment is done. In SQL Server you need to make sure that the default schema for the user is set in the database. It is recommended that for each user you set the same value for the schema in the database - if you do not set the default schema for each user then it would be defaulted to 'dbo' and all the components would get configured with that schema resulting in a non-working environment. The database panel should have values of the schema which correspond to that user. If there is no option to enter a schema value in the field its expected that the default schema which is the same as the user would be set in the database.

- e. Because the component requires manual steps to create the required tables, the Business Space Create Tables check boxes are disabled. Create the tables for this components by following step 6 on page 95.
 - f. Complete the rest of the steps to create the environment and save the settings. You can see **Servers > Deployment Environment** but the deployment environment is not started. Do not start the deployment environment at this time.
5. Optional: If you cleared **Create Tables** when you created the profile in 4 on page 94, generate the scripts for the message engine.
- a. In the administrative console, select **Servers > Deployment Environment > *your_deployment_environment* > Deferred Configuration**.
 - b. On the command line, go to where you want to generate the scripts.
 - c. Run `sibDDLGenerator.bat` utility to generate the scripts for each of the schemas required in your environment. For more information about running the utility, see the Deferred Configuration page. The schema names are the values you have chosen in the database panel above.

```
sibDDLGenerator.bat -system sqlserver -version 2005 -platform windows
-schema WPRCM00 -user user_name -statementend ; > output_script_filename
```

Remember to use the correct schema, which is listed in the Deferred Configuration page, and user name. Also, redirect the result to a file. Otherwise, the generated script is printed at the command prompt instead of in a file.

Note: If you configured the databases using a database design file, it is not necessary to run the `sibDDLGenerator.bat` utility. For more information, see “Creating database design files by using the database design tool” on page 97.

- 6. Manually create the Business Space database:
 - a. In the administrative console, select **Servers > Deployment Environment > *your_deployment_environment* > Deferred Configuration**.
 - b. Find the Business Space scripts.
 - c. Run the `createDatabase_BusinessSpace.sql` script and then the `createTable_BusinessSpace.sql` script.
- 7. In the administrative console, select **Servers > Deployment Environment > *your_deployment_environment* > Deferred Configuration** and click **Configuration Done**.
- 8. Log off of the administrative console, shut down the deployment manager, and then shut down all of the custom profiles.

9. Optional: Clean all applicable profile logs or save them in another directory. You may want to clean or move the logs as they will be appended with the last configuration. This can make it difficult to view the most current information.
10. Start the custom profiles, start the deployment manager, and then log in to the administrative console.
11. Start the deployment environment:
 - a. In the administrative console, start the deployment environment by clicking **Servers > Server Types > Deployment Environments**. Select the check box next to the deployment environment and clicking **Start**.
 - b. After 5 to 10 minutes (or longer, depending on the system), refresh the deployment environment page; the Status of the deployment environment changes to **started**.
12. Optional: Check the status of the following items:
 - a. In the administrative console, select **Applications > Enterprise Applications** and check that the installed applications started successfully.
 - b. Select **Resources > JDBC > Data sources** and test that the connection of every component that is not related to the message engine (that is, every component that does not include ME in the name) is successful.

Creating the Common database and tables after profile creation or augmentation

If you postponed creating the Common database and its tables by clearing the **Run database scripts to create database tables (do not select if using a remote database)** check box on the Database configuration panel in the Profile Management Tool, you or your database administrator must create the database and its tables manually. You can do this using scripts that the Profile Management Tool generates during profile creation or augmentation.

Before you begin

This topic assumes that you have performed the following actions:

- You created or augmented a stand-alone server or deployment manager profile using the Profile Management Tool
- In the Database configuration panel in the Profile Management Tool, you entered a name for the database in the **Common database name** or accepted the default common database name CMNDB
- In the Database configuration panel in the Profile Management Tool, you chose to delay creation of the Common database and its tables by clearing the **Run database scripts to create database tables** check box

About this task

Because an installation of WebSphere ESB requires the Common database to function, if you did not allow the Profile Management Tool to create it automatically, you or your database administrator must now create the database and its tables manually by using scripts that the Profile Management Tool generated during the profile creation or augmentation.

Procedure

1. Go to the directory containing the **configCommonDB.sh** script on Linux and UNIX platforms or the **configCommonDB.bat** script on Windows platforms. The default directory to which database scripts are output is:

- **Linux** **UNIX** `profile_root/dbscripts/CommonDB/db_type/db_name`
- **Windows** `profile_root\dbscripts\CommonDB\db_type\db_name`

Note: The Profile Management Tool provides an option to override the default directory. If you selected the option to override the default directory, the location to which database scripts are output is the path you entered in the **Database script output directory** field on the Database Configuration – Part 1 panel.

The variable `db_type` represents the supported database product and `db_name` represents the name of the database.

You must pass the **createDB** parameter to the `configCommonDB` script if you want to create a new local database; otherwise an existing database will be used.

Note: For Oracle, the batch file creates tables on an existing schema, so the **createDB** parameter should not be specified.

For example:

`configCommonDB.sh createDB` - creates the database and also the tables

`configCommonDB.sh` - creates only the tables and assumes that the database already exists

2. Use your standard database definition tools, native commands, and procedures to create the database and required tables by running this script. The script contains only basic statements for creating databases, tables, and indexes.

What to do next

After database creation completes successfully, before starting the server or deployment manager, be sure the database is running even if it is installed locally. Then start the server or deployment manager from the profile's First steps console to ensure there are no errors. You can check the `SystemOut.log` and `SystemErr.log` files for errors. These files are found in the following locations:

- `profile_root/logs/server_name`, for a stand-alone profile
- `profile_root/logs/dmgr`, for a deployment manager profile

Creating database design files by using the database design tool

Use the database design tool to create and generate a design of your database configuration. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of WebSphere ESB.

Creating a database design file for a stand-alone profile or deployment environment by using the database design tool:

You can use the database design tool to generate a design file for database tables that can be used by profile creation or when using the deployment environment wizard. The database design tool generates the design file from user-interactive input or from an existing design file.

Before you begin

Ensure that you have installed WebSphere ESB. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator (DBA) or solution architect. Alternatively, it might be a description of required parameters and properties.
- Information about how WebSphere ESB and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

Before you run the database design tool, ensure that you have made the following decisions:

- The type of deployment environment in which the database will be used (stand-alone profile or network deployment environment) based on scalability and high-availability requirements.
- The location of database tables.
- Details about the database type, specifically, but not limited to, the following items:
 - Type of database (DB2, Oracle, DB2 for z/OS, SQL Server)
 - Location of the JDBC driver on the system where the server profile will be created
 - User ID and password for authenticating to the database

Tip: Plan for database use when you review information about your planned usage of WebSphere ESB so that you make the necessary decisions on information needed by the database design tool.

About this task

This task describes how to use the database design tool to create a database design file for a stand-alone profile or deployment environment. The input for the database design tool is either user-interactive input or an existing design file. The available options change depending on your environment.

The **DbDesignGenerator** command has the following options.

```

-? -help
display help info.

-e db_design_file_name
edit the specified database design file (e.g., *.dbdesign, *.properties).

-r db_design_file [db_scripts_output_directory]
when a db_design_file is given, validation will be done on the specified
database design file based on the database type.
then a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template all generator can be validated.

-g db_design_file [-d output_directory] [-db_design_file2] [-d output_directory2] ...
[db_design_file2] [-d output_directory2]
generate the database scripts from the specified design files in batch mode.
the generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.

```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Procedure

1. Access the **DbDesignGenerator** command and run the file.

You can find the **DbDesignGenerator** command in the following location:

- **Windows** `install_root\util\dbUtils`

For example, **C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils>DbDesignGenerator.bat**

- **Linux** **UNIX** `/install_root/util/dbUtils`

For example, `/opt/IBM/WebSphere/AppServer/util/dbUtils>
DbDesignGenerator.sh`

Tip: If you see the message The system cannot find the specified path, you might have entered the path name incorrectly. Re-enter the path. When the database design tool launches successfully, you see the following information:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment  
(2)Create a database design for a single component  
(3)Edit an existing database design  
(4)Generate database scripts from a database design  
(5)exit [q]
```

```
Please enter the number for the design option :
```

2. To select the option (1)Create a database design for Standalone profile or Deployment Environment, type the number 1 and press Enter.

You are prompted to choose a database pattern; for example:

```
[info] Please pick one of the following [database pattern(s)] :
```

```
(1)bpm.advanced.nd.topology  
(2)bpm.advanced.standalone  
(3)wesb.nd.topology  
(4)wesb.standalone
```

3. To create a database design pattern for the stand-alone profile or deployment environment that you plan to configure, type the number for the appropriate option and press Enter. For a stand-alone profile, select options that include ".standalone;" for a deployment environment, select options that include ".nd."

For example, to configure the database pattern for a deployment environment for WebSphere Enterprise Service Bus, type the number 3 to select option (3)wesb.nd.topology, and press Enter. You see information similar to the following example:

```
[info] Please edit any database component with status of 'not complete' for required properties.
```

```
[info] Completed database components can be edited to change existing or defaulted property values.
```

```
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.
```

```
[info] Please pick one of the following [database component(s)] :
```

```
(1)[CommonDB] WBI_CommonDB : [master] [status = not complete]  
(2)[BSPACE] WBI_BSPACE : [status = not complete]  
(3)[SibME] WBI_CEI_ME : [status = not complete]  
(4)[SibME] WBI_SCA_APP_ME : [status = not complete]  
(5)[SibME] WBI_SCA_SYS_ME : [status = not complete]  
(6)[save and exit]
```

4. Type the number for the appropriate option to configure the master database component, and press Enter. You see the database components that can be configured for the previously selected environment. The database component listed as the master component lists [master] beside the name and must be configured first.

For example, to configure the master component for the (3)wesb.nd.topology design pattern, type the number 1 to select option (1)[CommonDB] WBI_CommonDB : [master] [status = not complete], and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
```

```
[ 1 ] CommonDB.WBI_CommonDB : : DbType key is not set.
```

```
Edit this database component? (y/n) [default=y] :
```

5. To edit the database component and select the database type that you are configuring, type `y` and press Enter.

After you choose to edit the database component, you see information similar to the following example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

6. Type the number that corresponds to the database type that you want to use for your environment, and press Enter. You obtain a set of prompts to specify the database properties. These prompts vary, depending on the database type that you plan to use.

For example, type the number 1 to select (1)DB2-distributed as the database type. After you select this database type for configuration of the Common DB, you see information similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :
```

```
[info] Please pick one of the following [Is this database for a Process Center?(s)] :
```

```
(1>false
(2>true
```

```
Please enter the number for the Is this database for a Process Center? [default=false] :1
The user ID you use for administrative security[default=] :
The password for the name specified with the adminUserName parameter[default=] :
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSpace] :
```

7. At each prompt, if a default value is listed, enter the appropriate value for your database configuration, or press Enter to accept the default value.

After you complete the last prompt for the database properties, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

To skip data source properties, enter `'s'`; or enter anything else to continue :

8. To configure the data source properties component, type anything other than `s` and press Enter. To skip this configuration and accept the defaults, type `s` and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you chose to configure the data source for your selected database type, you see the list of database providers for the data source. For example, you might see the following database providers for the DB2-distributed database type:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA)
(2)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for (1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA) as the database provider, type the number 1 and press Enter. After you select this database provider, you see information similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, enter the appropriate value for your database configuration, or if a default value is listed, press Enter to accept the default value.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):

-----

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components, since other components may inherit values from them.

[info] Please pick one of the following [database component(s)] :

(1)[CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BSPACE] WBI_BSPACE : [status = complete]
(3)[SibME] WBI_CEI_ME : [status = complete]
(4)[SibME] WBI_SCA_APP_ME : [status = complete]
(5)[SibME] WBI_SCA_SYS_ME : [status = complete]
(6)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as [status = complete] along with the master component. If this cannot be done for any reason, they remain listed as [status = not complete].

9. Optional: Follow the preceding steps to configure the remaining database components that are listed as [status = not complete]. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as [status = complete] as the result of configuring the master database component.
10. When all database components for your database pattern have been configured and are listed as [status = complete] in the database design tool, enter the appropriate number to select [save and exit], and press Enter. For example, after you finish configuring the (3)web.nd.topology database pattern, type the number 6 and press Enter. You see information similar to the following example:

```
[status] web.nd.topology is complete with 0 remaining item(s):

Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

11. Enter the location where you want to save the database design file, and press Enter. After you enter the location at the prompt, you see information similar to the following example:
Please enter the output filename [default=web.nd.topology.dbDesign] :
12. Enter the file name for the generated database design file, and press Enter. After you enter the file name at the prompt, you see information similar to the following example:
generate database scripts? (y/n) [default=y] :
13. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter.

- a. Specify the full path of the output directory that will contain the scripts for that database component, and press Enter.

After you type `y` and press Enter to indicate that you want to generate database scripts, you see information similar to the following example for each database component:

```
Please enter the output directory for WBI_CommonDB [default=DB2-distributed-CommonDB] :
```

After you type the location for the output directory and press Enter, you see information similar to the following example after each entry:

```
[info] The script(s) have been generated in C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB
```

After you enter the values for each prompt, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

Results

A database design file and, optionally, database scripts are created at the locations that you specified.

What to do next

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Creating a database design file for a specific component by using the database design tool:

You can use the database design tool to generate a design file for database tables required by specific components. The database design tool generates the design file from user-interactive input or from an existing design file.

Before you begin

Ensure that you have installed WebSphere ESB. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator (DBA) or solution architect. Alternatively, it might be a description of required parameters and properties.
- Information about how WebSphere ESB and its components have been installed, the database software used, and the properties required by that type of database.

- An understanding of the profiles you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

Before you run the database design tool, ensure that you have made following decisions:

- The type of deployment environment in which the database will be used (stand-alone profile or network deployment environment) based on scalability and high-availability requirements.
- The location of database tables.
- Details about the database type, specifically, but not limited to, the following items:
 - Type of database (DB2, Oracle, DB2 for zOS, SQL Server)
 - Location of the JDBC driver on the system where the server profile will be created
 - User ID and password for authenticating to the database

Tip: Plan for database use when you review information about your planned usage of WebSphere ESB so that you make the necessary decisions on information needed by the database design tool.

About this task

This task describes how to use the database design tool to create a database design file for a specific component. The input for the database design tool is either user-interactive input or an existing design file. The available options change depending on your environment.

The **DbDesignGenerator** command has the following options.

```
-? -help
display help info.

-e db_design_file_name
edit the specified database design file (e.g. *.dbDesign, *.properties).

-r db_design_file [ db_scripts_output_directory
when a db_design_file is given, validation will be done on the specified
database design file based on the database users.
then a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template all generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_file3] [-d output_directory3]
generate the database scripts from the specified design files in batch mode.
the generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

Restriction: The database design tool does not support Common Event Infrastructure (CEI).

Procedure

1. Access the **DbDesignGenerator** command and run the file.

You can find the **DbDesignGenerator** command in the following location:

- **Windows** `install_root\util\dbUtils`

For example, `C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils>DbDesignGenerator.bat`

- **Linux** **UNIX** `/install_root/util/dbUtils`

For example, `/opt/IBM/WebSphere/AppServer/util/dbUtils>DbDesignGenerator.sh`

Tip: If you see the message The system cannot find the specified path. you might have entered the path name incorrectly. Re-enter the path.
When the database design tool launches successfully, you see information similar to the following example:

```
[info] running DbDesignGenerator in interactive mode...
```

```
[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.
```

```
[info] To accept the given default values, simply press the 'Enter' key.
```

```
[info] Please pick one of the following [design option(s)] :
```

```
(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]
```

2. To select the option (2)Create a database design for a single component, type the number 2 and press Enter.

You are prompted for a component; for example:

```
[info] Please pick one of the following [component(s)] :
```

```
(1)bspace
(2)cei
(3)commondb
(4)sca
(5)sibme
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the Common database component, type the number 3 to select option (3)commondb, and press Enter. You see information similar to the following example:

```
[info] Please pick one of the following [database type(s)] :
```

```
(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

4. Type the number that corresponds to the database type that you want to use for your environment, and press Enter. You obtain a set of prompts to specify the database properties. The prompts vary, depending on the database type that you plan to use.

For example, type the number 1 to select (1)DB2-distributed as the database type. After you select this database type for configuration of the database, you see information similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :db2admin
Schema name[default=] :wesb
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

5. At each prompt, enter the appropriate value for your database configuration, or if a default value is listed, press Enter to accept the default value.

After you complete the last prompt, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.
```

```
To skip data source properties, enter 's'; or enter anything else to continue :
```

6. To configure the data source properties component, type anything other than s and press Enter. To skip this configuration and accept the defaults, type s and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you decided to configure the data source for a database after you selected DB2-distributed as your database type, you see information similar to the following example:

```
[info] Please pick one of the following [database provider(s)] :
```

```
(1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA)
(2)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

- a. Type the number for the appropriate option to select the database provider for the data source, and press Enter. For example, to select the option for (1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA) as the database provider, type the number 1 and press Enter. After you select this database provider for the data source, you see information similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

- b. At each prompt, if a default value is listed, press Enter to accept the default, or enter the appropriate value for your configuration.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

7. Enter the location where you want to save the database design file, and press Enter. After you enter the location, you see information similar to the following example:

```
Please enter the output filename [default=CommonDB_DB2-distributed.properties] :
```

8. Enter the file name for the generated database design file and press Enter. After you enter the file name, you see information similar to the following example:
generate database scripts? (y/n) [default=y] :

9. Optional: If you also want to generate database scripts based on the information provided to the database design tool, perform the following steps:

- a. Type y and press Enter.

After you type y and press Enter to indicate that you want to generate database scripts, you see information similar to the following example:

```
Please enter the output directory for CommonDB [default=DB2-distributed-CommonDB] :
```

- b. Specify the full path of the output directory that will contain the scripts for that database component, and press Enter.

After you enter the location for the output directory, you see information similar to the following example:

```
[info] The script(s) have been generated in C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-BPM_ProcessServer
```

After you enter the values for each prompt, you see information similar to the following example:

[info] thanks, quitting now ...

Results

A database design file is created and, optionally, database scripts are created at the location that you specified.

What to do next

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required `userName` and `password` properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):  
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.  
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-
```

```
...  
[WARNING] 2 potential problems are found in the scripts. They are  
DB_USER @ line 46 in file configCommonDB.bat  
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Creating and configuring the DB2 for z/OS database

If your deployment environment relies on a remote DB2 for z/OS database, use the procedures and reference information in this section to help you configure the database and create the database tables.

Related tasks:



Configuring an existing database during a typical installation

Use the information in this topic to determine the correct database values for configuring your existing database during a typical installation.

Create the DB2 for z/OS databases and storage groups using SPUFI, DSNTEP2, or DButility.sh:

The profile creation process generates Data Definition Language (DDL) scripts that you can use to create the DB2 database objects for the configuration. There are several tools that you can use to run the DDL scripts to create the database objects for your configuration. You can also use tools such as SPUFI or DSNTEP2 to create and populate the database.

Before you begin

Before you create the DB2 databases and storage groups, you must complete the following tasks:

- Create the server configuration. See *Roadmap: Installing and configuring IBM BPM Express* for information about how to create a configuration for a stand-alone server and network deployment environment.
- Make sure that the DDL has been generated for all the components you want to configure the database with. You can generate the DDL by completing the following tasks:

- Designing the database configuration

For a network-deployment environment, using the database design tool (DDT) is recommended.

For a stand-alone server environment, the database panels of the Profile Management Tool are usually enough to for stand-alone profiles, although you can use the DDT.

The output of the DDT is a design document (xml file) of the database configuration and, optionally, the SQL scripts to create the database tables.

- Prepare to use the DDL files

- You might need to copy the DDL files from the WebSphere ESB file system into a partitioned dataset (PDS). You can use a tool such as **Dd12Pds.sh** to copy the files.
- There is no restriction on the naming or organization conventions that apply to the database objects.
- The CEI DDL and the SIB DDL files need to be customized before they can be run.

Note: You can use the sample SIB DDL provided for single database configuration.

About this task

You can run the DDL scripts using **DBUtility.sh**, SPUFI, or DSNTEP2. You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 objects, particularly in a production environment. The tools can produce an audit trail of the DB2 database commands that have been issued.

If you want to create the database objects across multiple databases but still want to work in the USS environment, you can run the DDL scripts using the **DBUtility.sh** script several times specifying different components for each database name.

If you want to work in the USS environment, you can run the DDL scripts using the **DBUtility.sh** script, which is also supplied with WebSphere ESB.

Important: After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed width MVS data sets.

Procedure

1. Create the databases and storage groups.
2. Populate the databases using the generated DDL scripts. The location of the generated DDL scripts depends on how they were generated.

You can find the **DbDesignGenerator** command in the following location:

- **Windows** `install_root\util\dbUtils`

For example, `C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils>DbDesignGenerator.bat`

- **Linux** **UNIX** `/install_root/util/dbUtils`

For example, `/opt/IBM/WebSphere/AppServer/util/dbUtils>DbDesignGenerator.sh`

For DDL generated by other means, the DDL is in the directories under the following locations:

- `WAS_HOME/profiles/default/dbScripts` for a stand-alone configuration.
- `WAS_HOME/profiles/default/dbScripts` for a network deployment configuration.

Where `WAS_HOME` is the top directory of your WebSphere Application Server configuration.

3. If you are running the DDL from a USS environment, assign the appropriate permissions to the copies of the files; for example:
`chmod 755 createTable_AppScheduler.sql`
4. Edit the values in the file to suit your needs. The database names, storage groups and schema names are customized by the product configuration process. Check the values in each file to make sure they match the values that you entered in the response file that provided input to the configuration script and are suitable for your database.

Note: The files can be provided in ASCII format. If the tools that you use to view, edit, and run the scripts require the scripts to be in EBCDIC format, use the **iconv** command to convert the file to EBCDIC. For example:

```
iconv -t IBM-1047 -f ISO8859-1 createTable_AppScheduler.sql >
createTable_AppScheduler_EBCDIC.sql
```

If you have converted the file from ASCII format to EBCDIC but need to run the file in ASCII format, use **iconv** to convert the file back to ASCII. For example:

```
iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
createTable_AppScheduler.sql
```

5. Optional: If you want to create database objects outside of the USS environment, for example, by using SPUFI or DSNTEP2, you can use the supplied Ddl2Pds.sh script to copy the customized DDL from USS to a partitioned dataset. For example, to copy the DDL for the WebSphere ESB Common component, enter a command similar to the following from the /usr/lpp/zWESB/V7R5M1/zos.config/samples directory:

```
./Ddl2Pds.sh -Source
/WebSphere/V7S05Z1/AppServer/profiles/default/dbscripts/CommonDB/DB2zOS/S5CELLDB -PDS HEALDR.DDL2PDS.TEST -Component
WPS
```

6. Run the customized scripts using the tool of your choice. For example:

SPUFI A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

DSNTEP2

A sample dynamic SQL program provided with the DB2 for z/OS product.

DBUtility.sh

DBUtility.sh is a utility that is supplied with WebSphere ESB for z/OS and installed in the installation file system. For example:

/usr/lpp/zWESB/V7R5M1/bin/DBUtility.sh. You can use this utility to create the database and storage groups, as well as to run the SQL to create the database tables later, from USS. **DBUtility.sh** uses ASCII input. Here is an example of the syntax used with the **DBUtility.sh** script:

```
/WebSphere/V7S03Z1/AppServer/profiles/default/bin/DBUtility.sh
createTable
-DdbStorageGroup=S3DBST0
-DdbSchemaName=S3CELL
-DsqlScriptName.default=createTable_AppScheduler.sql
-DsqlScriptPath.default=/WebSphere/V7S03Z1/AppServer/profiles/default/dbscripts/CommonDB/DB2zOS/S3CELLDB
/createTable_AppScheduler.sql
-DdbType=DB2UDBOS390
-DdbName=S3CELLDB
-DprofileName=default
-DprofilePath=/WebSphere/V7S03Z1/AppServer/profiles/default
-DdbJDBCProperties=/wps/dbscripts/db2v9
-DdbConnectionLocation=DSN810PP
-DdbJDBCClasspath=/usr/lpp/db2910/db2910/jcc/classes
-DdbUserId=wsadmin
-DdbPassword=password
-DdbDelayConfig=false
-DdbCreateNew=false
-DdbHostName=winmvsp1.hursley.ibm.com
-DdbServerPort=448
>/tmp/output.out 2>/tmp/error.out
```

7. Verify that the database, storage group, and tables have been created successfully with no errors by inspecting the output.
8. If you are creating a stand-alone configuration, verify the WebSphere ESB installation:
 - a. Start the server.
 - b. Open the administrative console by opening a browser window and typing the URL of the server that you want to view. For example:
http://server_name.domain_name:port_number/admin
 - c. Log in to the administrative console.
 - d. Verify that you can see WebSphere ESB for z/OS on the Welcome page. You can click it for more information.
 - e. Navigate around the console to check that the server has a status of started. Also check that all the applications are started, and that the messaging

engines are started. If anything has failed to start, you can look in the server job logs for "SEVERE" or "WARNING" messages that provide details about the failure.

Results

The DB2 databases and storage groups are created and populated with the necessary database objects, such as tables and indexes.

What to do next

If you are creating a stand-alone configuration, you can now deploy applications to the server.

If you are creating a network deployment configuration, you must create one or more empty nodes to add to the deployment manager cell. See *Configuring the software after a Custom installation to create one Deployment manager and Custom profiles*.

Granting table privileges to the JCA authentication alias user ID:

If the schema name you are using is not the same as the JCA authentication alias user ID you must grant a sub-set of DB2 privileges to the JCA authentication alias user ID.

About this task

The DDL for the Service Integration Bus already contains commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other WebSphere ESB components do not supply any GRANT statements.

Use a schema name that is not the same as the JCA authentication alias to prevent the alias user ID having the power to drop tables. (The power to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to DROP tables.

If you want the WebSphere ESB to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the DDL and editing it to construct GRANT commands from the CREATE commands. Create GRANT commands like:

```
GRANT ALL PRIVILEGES ON TABLE  
cell.tablename TO userid/sqlid
```

Where *userid/sqlid* is the JCA authentication alias user ID.

Setting the correct schema name for the SIBs:

To ensure the SIB messaging engines can access the appropriate DB2 tables, set the correct schema name for the SIB messaging tables to use to access the DB2 tables.

Before you begin

Start the server (stand-alone server or deployment manager).

About this task

Use the administrative console to change the schema names.

Procedure

1. Log in to the administrative console.
2. Navigate to **Service Integration > Buses**.
3. For each bus:
 - a. Select **Messaging engines**, then click the name that is displayed.
 - b. Click **Message store**.
 - c. Change the value of **Schema name** to the name used when creating the DB2 tables for this SIB.
 - d. Click **Apply**.
 - e. Save your configuration changes.
4. Log out of the administrative console.
5. Stop, then restart the server.
6. Look in the output of the Adjunct job log for successful SIB messaging engine startup messages. For example:
BB000222I: "BusName"
CWSID0016I: Messaging engine MessagingEngineName is in state Started.

Results

The schema name used by the SIB messaging tables to access the DB2 tables is changed.

Verifying the installation with DB2 for z/OS:

When verifying an installation with a DB2 for z/OS database, it is important to check the Servant and Adjunct job logs to see whether there are any error messages that might indicate problems accessing the data store.

Procedure

1. Ask your DB2 system administrator to check the authorities that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration.
2. Ask your DB2 system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over to people to use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.
3. Log in to the administrative console.
4. In the administrative console, check that all the applications are started, the messaging engines are started, and all the data sources can be accessed using the **Test Connection** option. If any application has failed to start, look in the Servant and Adjunct job logs for SEVERE or WARNING messages that provide detail about the failure.
 - If you see DB2 errors such as SQLCODE -204, in the administrative console, set the correct schema name or currentSQLID value in the custom properties

section of the data sources. If the schema name is not the same as the user ID in the JCA authentication aliases, the SQL requests try to find tables qualified by the user ID in the JCA authentication alias.

- If you see DB2 deadlock errors such as SQLCODE -913 Reason Code 00C90088, set the RRULOCK DB2 parameter to YES to prevent tablespace locks on WebSphere ESB tables.

What to do next

If all the messaging engines have initialized correctly, and you do not see any other errors related to opening JDBC connections, you can continue to customize your configuration of WebSphere ESB.

Modifying the transaction log options for a DB2 database

When you configure DB2 for use with WebSphere ESB, you must modify the transaction log options.

Procedure

1. Start a DB2 command line processor.
2. Run the following commands:

```
CONNECT TO [DB_name]
UPDATE DB CFG FOR DB_name USING LOGFILSIZ 4096 IMMEDIATE
UPDATE DB CFG FOR DB_name USING LOGSECOND 64 IMMEDIATE
CONNECT RESET
```
3. Stop and restart DB2.

Configuring a network deployment environment

After you have finished installation, you can build a network deployment configuration.

Creating a network deployment environment

After performing a Custom installation, you can use the Profile Management Tool or the **manageprofiles** command-line utility to build a network deployment cell across multiple machines. For each cell, you create a deployment manager profile and one or more custom (managed node) profiles. You can then create the deployment configuration that you want.

The information in this section describes how to configure a network deployment environment; in summary:

1. Create a deployment manager profile and one or more custom (managed node) profiles, by using the Profile Management Tool
2. Create a deployment configuration, by using either the Deployment Environment wizard or the administrative console.

Tip: Instead of using the Profile Management Tool, you can use the **manageprofiles** command-line utility to create profiles, as described in Configuring the software using command-line utilities and **wsadmin**.

Configuring a network deployment environment using the deployment environment wizard:

After performing a Custom installation and creating the deployment manager and custom (managed node) profiles, you can create a network deployment configuration.

The information in this section describes how to use the deployment environment wizard to create a network deployment environment based on the topology pattern templates packaged with the software.

The information in this section assumes that you have run the installer and have performed a Custom installation and that you have created the deployment manager and custom (managed node) profiles.

For information about using wsadmin to create a network deployment configuration, see *Creating deployment environments using the command line*.

Creating a deployment environment from a pattern:

After determining the pattern on which to base your network deployment configuration, use the Deployment Environment wizard to create the deployment environment that is based on the pattern.

Before you begin

Tip: An alternative to using a pattern to create the deployment environment is using an imported design. For more information, see Importing deployment environment definitions based on design documents using the administrative console

On the administrative console of the deployment manager navigate to **Servers > Deployment Environments**.

Required security role for this task: When security and role-based authorization are enabled, you must log in to the administrative console as an administrator or configurator to perform this task.

You should have planned the patterns and features that you want to configure. For more information, see Planning a network deployment environment.

It is assumed that you have installed the product and that you have created the deployment manager profile and the associated nodes.

Additionally, one of the steps in the Deployment Environment Configuration wizard includes importing a database design document. The database design document defines the database configuration for the selected deployment environment features. WebSphere ESB includes a response-driven database design tool (DDT) that creates a database design document based on user inputs. In addition to creating the design document, the DDT provides an option to create the database tables as well. Typically, the person running the DDT would choose to create the database tables at that point. The document can then be used by the WebSphere ESB deployment environment wizard to configure the databases used in the deployment environment. For more information about the DDT and database configuration in general, see Configuring databases.

About this task

This task describes the procedure for creating a deployment environment that is based on a specific pattern and uses the Deployment Environment Configuration wizard.

Note: If you make an error while you are working in the wizard, you can go back by clicking **Back**.

Procedure

1. From the administrative console, go to the Deployment Environments page. For example, click **Servers > Deployment Environments**
2. Launch the Deployment Environment Configuration wizard by clicking **New** on the Deployment Environments page.

- a. The **Create a deployment environment based on a pattern** option is selected. **Create a deployment environment based on a pattern** is the system default and it is the option described in this topic.

Deployment environment patterns capture commonly used business integration topologies. A pattern provides you with a template for the deployment environment that you are creating.

Note: Patterns have a direct relationship to the products supported by the configured deployment manager. WebSphere ESB supports a specific set of patterns, with the *Remote messaging and remote support* pattern being the system default. If your deployment manager supports other products in addition to WebSphere ESB, additional patterns may apply. Consult product-specific documentation for information on patterns as they apply to the products.

For information on the types of patterns provided with WebSphere ESB, see *Topology types and deployment environment patterns*.

- b. Enter a unique name for the deployment environment in the **Deployment environment name** field.
- c. Optional: To view all of the configuration steps in the wizard, select **Detailed: Show all steps**.

If you choose **Fast path: Show only needed steps** the wizard displays only those pages that **do not** have assigned default values. Choose **Fast path: Show only needed steps** only if you are agreeable to accepting the system-provided default values for the deployment environment configuration.

This topic assumes that you have chosen **Detailed: Show all steps**

- d. Click **Next** to display the Deployment Environment Features page.
3. On the Deployment Environment Features page, select the feature for the deployment environment and click **Next** to either view a list of compatible features, or to view a list of deployment environment patterns. Features represent the runtime processing capabilities of your deployment environment.

The list of available features on the Deployment Environment Features page is based on the deployment manager profile. If your deployment manager profile has been augmented to include other products alongside WebSphere ESB (for example, IBM Business Monitor), then the Deployment Environment Features page also lists these features.

If you have installed and configured a profile for WebSphere ESB, then the Deployment Environment Features page includes the following details:

WESB, for WebSphere Enterprise Service Bus, which provides a deployment environment that supports mediations.

The default value for the deployment environment feature matches the runtime capabilities of your deployment manager.

4. On the Select compatible deployment environment features page, select additional features as necessary and click **Next** to view the list of patterns associated with your primary and ancillary feature selections.

Note: The Select compatible deployment environment features page is displayed only if the deployment manager has been augmented with other business process management (BPM) features, such as IBM Business Monitor.

For an understanding of the relationship of features and compatible features, see the information on deployment environments in the Planning section.

5. On the Select the deployment environment pattern page, select the pattern for the selected deployment environment, then click **Next** to display the Select Nodes page.

The list of patterns that display on the Deployment Environment Patterns page is dynamic. This list is activated by, and dependent on, the following environment conditions and configuration decisions:

- The platform on which you have installed the software
- The selections that you have made on the Select the deployment environment feature page and the Select compatible deployment environment features page.

For a detailed description of the relationship of patterns to features, see Topology patterns and supported product features

6. Optional: On the Select Nodes page, select the nodes to be included in the deployment environment then click **Next** to display the Clusters page.

Select nodes that have the required capabilities for the environment you selected in step 3 on page 223. For example, if you selected **BPMSPC** as your Deployment Environment type, the nodes selected should address the capabilities of that environment type.

Select at least one node for the deployment environment. For high-availability and failover environments, select at least two nodes. For scalability, select all nodes.

To include a node, select the check box next to the node name. Use **Node Mapping** to map the selected node to another node name.

7. Optional: On the Clusters page, assign the required number of cluster members on each node for each cluster *type* (Application Deployment Target, Messaging Infrastructure and Supporting Infrastructure) of the deployment environment.

By default one cluster member is assigned on each node for each function. You change the number by replacing the number in each column. If you are unfamiliar with the different cluster roles and functions provided by each type of cluster, see “Topology types and deployment environment patterns.”

A 0 (zero) value for a node means that the node does not contribute to the selected function, based on features that you have selected.

After assigning cluster members, you can click **Next** to display the Cluster naming pages for each cluster type of the deployment environment. The Cluster naming sub-steps that display will vary depending on the deployment environment pattern selected.

The system generates default values for cluster names and cluster member names.

If you do not want to customize cluster names or cluster member names, you can use the wizard navigation pane to go directly to the REST Services page in a following step.

Each substep page is structured in the same fashion, and is described in Customize the cluster names and cluster member names.

- a. Optional: Customize the cluster names and cluster member names.

Use the Cluster Naming page to customize cluster names or cluster member names for the cluster type. There is one substep page for each cluster *type* in the pattern that you have selected. For example, if you selected a **Remote messaging and remote support pattern**, there are 3 sub-steps, one for each type of cluster (Application Deployment Target, Messaging Infrastructure and Supporting Infrastructure) in that pattern. The information on each substep page is as follows:

Cluster

A read-only field specifying the functional role of the cluster.

The value varies depending on the cluster type, as follows:

- Application Deployment Target
- Messaging Infrastructure
- Supporting Infrastructure
- Web Application Infrastructure

For information on the functional role provided by each cluster type, see Topology types and deployment environment patterns

Cluster Name

Contains the system-generated default value for the cluster name.

The default values are based on a naming convention of <Deployment Environment Name>.<Cluster type name>, where cluster type name is one of the following values:

- AppTarget
For clusters performing the role of application deployment target
- Messaging
For clusters performing the role of messaging infrastructure
- Support
For clusters performing the role of supporting infrastructure
- Web
For clusters performing the role of supporting web applications.

Note: This cluster type name applies for BPM configurations in which WebSphere Business Monitor is the primary feature / product.

Cluster Member Name

Accept the system-generated default value or specify a name of your choosing.

The default value for the cluster member name is based on the following naming convention: <cluster name>.<node name>.<node number sequence> .

The number of cluster member names that display in the table match the number of cluster members that you entered for the cluster type column and node row on the Clusters page. See the preceding step for the Clusters page.

8. On the REST Services page, configure service endpoints for Representational State Transfer (REST) application programming interfaces (APIs).

If you want widgets to be available in Business Space, you must configure the REST service endpoints for those widgets.

- a. Configure a full URL path for all REST services by selecting either **https://** or **http://** from the **Protocol** list.
 - b. Enter a name in the **Host Name or Virtual Host in a Load-Balanced Environment** field.
 - c. In the **Port** field, enter the port that a client needs to communicate with the server or cluster.
 - d. In the table of REST services, if you want to modify the description of the REST service endpoint, overwrite the entry in the Description field. The other fields are read-only.
 - e. Click **Next** to go to the Import the database configuration page.
9. Optional: On the Import the database configuration page, click **Browse** to go to the database design document or enter the path to the database design document and then click **Next** to go to the Data sources page. The design document can be based on a database design that you created using the database design tool (DDT), or it can be the supplied design document based on the pattern and feature that you have selected.

Note: The database design document that you import for the deployment environment does not change the commonDB created at Profile Creation time.

10. Required: On the Database page, configure the database parameters for data sources of the deployment environment, then click **Next** to go to the Security page.

Note: The database specified in this panel must already exist. Deployment environment configuration never creates a new database.

On this page, define the database information for the components that are included in this deployment environment. Where possible, the wizard supplies default information for the parameters, but change those values to match the values that you defined when you planned the environment.

Note: If you imported a database design document, the information on the Database page reflects the data source configuration as it exists in the database design document that you imported.

Whether or not this step displays for a fast path deployment environment configuration is conditional. This step displays for a fast path deployment environment configuration if more than one database has been defined.

This step always displays if you are using DB2 for z/OS or an Oracle database provider.

Note: The default schema names that are displayed on this page might conflict with your site naming convention or might conflict with existing schemas. As such, it is likely that you will need to change the schema name. Pay close attention to the values specified to avoid potential naming conflicts.

Oracle database considerations:

- Make sure that the username and schema name are exactly the same. The user specified should exist in the database before generating the environment.

SQL Server considerations:

- Make sure that the username and schema exist before the configuration is done. The schema value should be the default schema for the user chosen.

- To indicate that users will connect to the databases using Windows credentials, select the individual data source, click **Edit**, and select **Apply Windows authentication**.

For a production environment, you should set the same values for **User name** and **Schema name** and you should deselect **Create tables**. For a production environment, create the required schemas manually and use the SQL files generated to create the tables.

Note: You cannot select **Create tables** for Business Space (the option is unavailable for selection). The SQL files for Business Space need to be run manually. For information on running the SQL manually for Business Space, see Configuring Business Space database tables.

You can edit all key parameters, such as the database name, whether or not to create tables, the data source runtime user name, and the password for the deployment environment.

You can select which database to use for the given component.

DB2 for z/OS: The **Create tables** option cannot be used if you are using a DB2 for z/OS database provider.

Steps that cannot be completed through the Deployment Environment Configuration wizard, and which need to be completed manually, are listed on the Deferred Configuration page.

11. On the Security page, configure the authentication aliases WebSphere uses when accessing secure components

You can change the authentication alias user name and password on this page. These aliases are used to access secure components but do not provide access to data sources

12. Optional: On the System web applications page, set the context root for component-based web applications in your deployment environment or accept the system-provided default values for the context roots. Then click **Next** to display the Summary page.

The System web applications page displays for deployment environments using the Remote messaging, support and web applications pattern.

The table contains the following control information.

Web Application

The name of the Web application.

Some of the components that are part of the deployment environment you are creating contain web applications. The **Web application** column can include the following components:

- Business Space

Context Root

The current value of the context root for the component.

By default, the default context root for the web application applies. You can change the context roots by typing over the value in the **Context Root** field.

Note: The Business Space context root is read only and cannot be edited.

13. Verify that the information on the Summary page is correct and perform the following substeps:
 - a. Optional: If you do not want to save the deployment environment configuration, you can click **Cancel**.

- b. Optional: If you want to exit without generating the configuration, click **Finish**.
To get back to the panel (if you exited without completing), perform the following from the administrative console: **Deployment Environments** > *name of deployment environment* > **Generate Environment** .
- c. To save the deployment environment configuration, click **Finish** and from within the Messages window, click **Save**.
Clicking **Save** saves the deployment environment to the master configuration. If an error occurs during deployment environment generation, the configuration settings are saved to the master configuration.
- d. Check for deferred configuration steps
Select **Deployment Environments** > *name of deployment environment* > **Deferred Configuration**
You need to address any existing deferred configuration steps before starting the Deployment Environment.
- e. If you are satisfied with the deployment environment configuration and you have addressed any of the deferred configuration steps, click **Finish and Generate Environment** to save and complete the configuration of the deployment environment.

Results

When the configuration completes, you can examine the configuration files to view the changes.

What to do next

Either save the changes to the master configuration or discard them.

Configuring a network deployment environment using the administrative console:

After you perform a custom installation and create the deployment manager and custom (managed node) profiles, you can create a network deployment environment using the administrative console.

The information in this section describes how to use the administrative console to create a network deployment configuration.

This section assumes that the following information is true:

- You have run the installer to create a Custom installation.
- You have created the deployment manager and custom (managed node) profiles.
- You are familiar with network deployment topologies and configurations and the administrative console.

Important: Consider using the Deployment Environment wizard to create your network deployment environment.

Any operation that you can perform from the administrative console can also be performed with wsadmin. Additionally, command assistance is available for a subset of administrative console actions. When available, command assistance

displays the wsadmin scripting command for the last console action that you performed. For information about command assistance, see Administrative console actions with command assistance.

Creating and configuring servers and clusters:

You can use the administrative console to create and configure the servers and clusters for WebSphere ESB.

The information in this section describes how to create servers and clusters for WebSphere ESB manually using the administrative console.

Creating a cluster:

The following instructions explain how to create a cluster with one cluster member. The benefit of using the administrative console to create a cluster is that you can undo your changes as you go and you can use a graphical user interface.

Before you begin

Before you create a cluster using the administrative console, start the deployment manager.

About this task

The following procedure describes how to create a cluster with one cluster member using the administrative console. Note that the tasks performed from the administrative console that are described in this topic can also be performed using administrative scripting. For information on the **createCluster** parameters, see ClusterConfigCommands command group for the AdminTask object in the WebSphere Application Server information center.

For information on using the wsadmin tool to create cluster members, see Creating cluster members using scripting.

Consider to create a cluster even if a single server is currently sufficient for your high availability and scalability requirements because it is easier to add more servers to the cluster later.

Procedure

1. Log in to the administrative console and navigate to **Servers > Clusters > WebSphere application server clusters**.
2. Click **New** to display the Create new cluster page.
3. From the Create new cluster page, enter basic cluster information:
 - a. Type a name for the cluster in the **Cluster name** field.
 - b. Select **Prefer local** if you want to enable host-scoped routing optimization. This option is enabled by default. When this option is enabled, if possible, EJB requests are routed to the client host. This option improves performance because client requests are sent to local enterprise beans.

Note: If you enable the preferLocal optimization, the deployment manager must be running to affect the configuration. If the deployment manager is shut down, preferLocal optimization is not performed and requests might be dispersed across all the members of the cluster

- c. Select **Configure HTTP session memory-to-memory** replication if you want a memory-to-memory replication domain created for this cluster.
The replication domain is given the same name as the cluster and is configured with the default settings for a replication domain. When the default settings are in effect, a single replica is created for each piece of data and encryption is disabled. Also, the Web container for each cluster member is configured for memory-to-memory replication.
To change these settings for the replication domain, click **Environment > Replication domains > replication_domain_name**. To modify the Web container settings, click **Servers > Clusters > WebSphere application server clusters > cluster_name > Clusters members > cluster_member_name**. Then, in the **Container settings** section, click **Web container settings > > Web container > Session management > Distributed environment** settings in the administrative console. If you change these settings for one cluster member, you might also need to change them for the other members of this cluster.
4. Click **Next** to go to the Create first cluster member page.
5. From the Create first cluster member page, enter information about the cluster member.
 - a. Enter the member name in the **Member name** field
 - a. From the **Select node** list, click the node in which you want to define the server.
 - b. In the **Weight** field, enter the weight value for the cluster member.
The weight value controls the amount of work that is directed to the application server. If the weight value for this server is greater than the weight values that are assigned to other servers in the cluster, then this server receives a larger share of the workload. The weight value represents a relative proportion of the workload that is assigned to a particular application server. The value can range from 0 to 20.
 - On a **z/OS system** weight is used to balance some of the workload types, but others are balanced by the z/OS system. For HTTP requests, weights are used to distribute HTTP traffic between the Web server plug-in and the controller handling the clustered application server. Assign a higher weight value to the application server that should receive the HTTP traffic.
 - For Web services calls, information is transferred from a servant in one application server to a controller in another application server. The application server that receives the call has the highest weight value.
 - Weight has no affect on Internet Inter-ORB Protocol (IIOP) requests. IIOP requests are distributed to the correct application server using the sysplex distributor.
 - c. Select **Generate unique HTTP ports** (the default option) if you want to generate unique port numbers for every HTTP transport that is defined in the source server.
When this option is selected, this cluster member does not have HTTP transports or HTTP transport channels that conflict with any of the other servers that are defined on the same node. If you clear the **Generate unique HTTP ports** check box, all of the cluster members will share the same HTTP ports.
6. From the section, **Select basis for first cluster member**, select from the following options:
 - Create the member using an application server template

This is the typical way of creating a cluster. Select **defaultESBServer**.

If you select this option and click **Next**, a blank form is displayed which you can use to define additional cluster members. The server you have just created is listed at the bottom of the screen.

- a. Click **Next**.
- b. Check the details on the summary screen and click **Next**.
- c. Save your configuration changes.

The cluster you have just created is displayed in the list.

- d. Customize the port numbers to suit your configuration.
- Create the member using an existing application server as a template
This option is not supported.
 - Create the member by converting an existing application server

Note: Do not select this option. WebSphere Application Server added a new parameter named **resourcesScope** to the `createCluster` and `createClusterMember` commands that allows you to control how the resources will be processed when the first cluster member is added to a cluster. Those parameters are described in the technote titled `New parameter named resourcesScope was added to the createCluster and createClusterMember commands`.

It is required to use the **resourcesScope** parameter with the value `cluster` for WebSphere ESB. The **resourcesScope** parameter is not currently exposed in the administrative console. Using the administrative console to create a cluster using an existing single server as the first cluster member is not supported on WebSphere ESB. Selecting this option will result in the following error message in the administrative console or the `SystemOut.log` of the deployment manager:

`resourcesScope parameter needs to be set to 'cluster' if creating a cluster from an existing server`

Using the command line to create a cluster using an existing single server as the first cluster member is valid. You must set the value of the **resourcesScope** to `Cluster`. For example,

```
$AdminTask createCluster {-clusterConfig  
{-clusterName newcluster -preferLocal true}  
-convertServer {-serverNode NODE1Node01 -serverName testserver  
-resourcesScope cluster}}
```

For information on the **createCluster** parameters, see `ClusterConfigCommands` command group for the `AdminTask` object in the WebSphere Application Server information center.

For information on using the `wsadmin` tool to create cluster members, see `Creating cluster members using scripting`.

- None
Works always.

7. Click **Next**.
8. Create additional cluster members.

Before you create additional cluster members, check the configuration settings of the first cluster member. These settings are displayed at the bottom of the `Create additional cluster members` panel of the `Create a new cluster` wizard. For each additional member that you want to create:

- a. Specify a unique name for the member. The name must be unique within the node.

- b. Select the node to which you want to assign the cluster member.
 - c. Specify the weight you want given to this member. The weight value controls the amount of work that is directed to the application server. If the weight value for the server is greater than the weight values that are assigned to other servers in the cluster, then the server receives a larger share of the workload. The value can range from 0 to 20.
 - d. Select **Generate unique HTTP ports** if you want to generate unique port numbers for every HTTP transport that is defined in the source server.
 - e. Click **Add member**. You can edit the configuration settings of any of the newly created cluster members other than the first cluster member, or you can create additional cluster members. Click **Previous** to edit the properties of the first cluster member. The settings for the first cluster member become the settings for the cluster member template that is automatically created when you create the first cluster member.
9. When you finish creating cluster members, click **Next**.
 10. View the summary of the cluster and then click **Finish** to create the cluster, click **Previous** to return to the previous wizard panel and change the cluster, or click **Cancel** to exit the wizard without creating the cluster.
 11. To further configure a cluster, click **Servers > Clusters > WebSphere application server clusters**, and then click the name of the cluster. Only the **Configuration** and **Local Topology** tabs display until you save your changes.
 12. Click **Review** to review your cluster configuration settings. Repeat the previous step if you need to make additional configuration changes.
 13. If you do not want to make any additional configuration changes, select **Synchronize changes with Nodes** and then click **Save**. Your changes are saved and synchronized across all of your nodes.

Note: If you click **Save**, but do not select **Synchronize changes with Nodes**, when you restart the cluster, the product does not start the cluster servers because it cannot find them on the node. If you want to always synchronize your configuration changes across your nodes, you can select **Synchronize changes with Nodes** as one of your console preferences.

14. Restart the cluster.

Results

The cluster is created with your chosen server in the selected managed node as the first cluster member.

Creating a new server:

Most installations require several servers to handle the application serving needs of the production environment. You can use the command-line tool or the administrative console to create the servers you need.

Before you begin

Determine if you want to include the new server in a cluster. If this server is going to be part of a cluster, you must create the server with the **Create a new cluster** wizard instead of the **Create a new application server** wizard.

About this task

Important: This task creates a managed server. If you want a stand-alone server, do not follow these steps. Instead, create a stand-alone server profile.

To create a new managed server, perform the following steps.

Procedure

Follow the instructions in *Creating application servers*, selecting the **defaultESBServer** template or a suitable user-defined template from the *Select a server template* page. **Restriction:** The “Start components as needed” capability is not supported.

What to do next

Configure the components you need on the server. See **Configuring components** for more information.

Creating deployment environments using the command line:

You can use **wsadmin** to create a deployment environment. The **createDeploymentEnvDef** and **generateDeploymentEnv** provide a command-line equivalent to creating the deployment environment using the deployment environment wizard.

Creating deployment environment definitions using the command line:

You can create a deployment environment definition using the **wsadmin** command. Running **createDeploymentEnvDef** provides the definition of the deployment environment.

Before you begin

You must be on the deployment manager to create the deployment environment definition.

Required security role for this task: When security and role-based authorization are enabled, you must use a userid and password with administrator or operator authority to perform this task.

WebSphere ESB supports a set of patterns, *Remote messaging and remote support* being the pattern to use for a network deployment production environment. If your deployment manager supports other products in addition to WebSphere ESB, you might want to use patterns for those products when creating a deployment environment definition. For information about patterns for other products, see the documentation for those products.

About this task

This task creates a deployment environment definition that is based on a specific pattern and uses the **wsadmin** command.

You can use the **wsadmin** command to create the same deployment environment as you can create from the administrative console. This capability allows you to run the administrative task to create a deployment environment definition with all the

default values based on an existing configuration (the configuration that you created at profile creation time). The command also includes an optional property that imports a database design document. The database design document holds the database configuration for the topology you are creating.

When you generate the deployment environment, the information about whether to create tables is taken from the design document. Make sure that the **createTables** parameter is correctly set in the database design file that you specify.

Important: If you use the **createTables** parameter, the databases must already exist. Do not use **createTables** for a production environment where you want to customize the generated database scripts. Do not use **createTables** if you have a remote database server.

For Oracle databases, make sure that the user name and schema name are exactly the same. The user must exist in the database before you generate the environment.

For SQL Server databases, make sure that the user name and schema exist before the configuration is done. The schema value must be the default schema for the user chosen.

For a production environment, set the same values for user name and schema name and do not select **createTables**. Create the required schemas manually and use the generated SQL files to create the tables.

A *deployment environment definition* describes the specific component, configuration (of clusters, nodes, and servers), resources and related configuration parameters that make up a deployment environment. This definition can also be referred to as an instance of a deployment environment configuration. A deployment environment configuration can be exported into a deployment environment definition. You can import a deployment environment definition to add a new deployment environment configuration to your system.

Procedure

1. Open a command window. The **wsadmin** command can be found in either of the following directories:

```
install_root/profiles/dmgr profile/bin  
install_root/bin
```
2. At the command prompt, enter the **wsadmin** command to enter the **wsadmin** environment.
3. Use the **createDeploymentEnvDef** command to create the deployment environment definition with a specific name for a particular runtime and pattern.

Note: If administrative security is on, and you did not supply a user ID and password in the command, you are prompted for a user ID and password.

Example

This example creates a deployment environment definition for a remote messaging and remote support pattern on the WebSphere ESB runtime, with myDepEnv on the host myDmgr with administrative security enabled. The example imports a database design document named nd.topology.dbDesign:

```
wsadmin -connType SOAP -host myDmgr -port 8879
> $AdminTask createDeploymentEnvDef {-topologyName topOne
-topologyPattern RemoteMessagingAndSupport
-topologyRuntime WESB -dbDesign C:\dbDesigns\nd.topology.dbDesign}
> $AdminConfig save
```

Note: If you disable administrative security, you do not need to provide a user ID and password.

What to do next

After you have imported or created a deployment environment on a deployment manager, you can configure the deployment environment using the **generateDeploymentEnv** command.

Adding nodes to a deployment environment definition using the command line:

You can add nodes to a deployment environment definition using the **wsadmin** command.

Before you begin

The task assumes that the node has been federated to the deployment manager.

This command to add a node to the deployment environment definition will fail if the topology is already configured.

You must be on the deployment manager to add nodes.

Required security role for this task: When security and role-based authorization are enabled, you must use a userid and password with administrator or operator authority to perform this task.

About this task

This task adds a federated node to a deployment environment definition and uses the **wsadmin** command.

Procedure

1. Open a command window. The **wsadmin** command can be found in either of the following directories:
`install_root/profiles/dmgr profile/bin`
`install_root/bin`
2. At the command prompt, enter the **wsadmin** command to enter the **wsadmin** environment.
3. Enter the **addNodeToDeploymentEnvDef** command to add the node to the deployment environment definition.

Note: If administrative security is on, you will be prompted for a user ID and password, if you do not supply it in the command.

Example

This example adds a node (**MyNode**) to deployment environment definition (**myDepEnv**) with administrative security enabled:

Attention: If you are adding a node to a single cluster topology pattern, the value for `-topologyRole` must be set to **ADT**. Deployment environment topology patterns are specified when you create the deployment environment using either the `createDeploymentEnvDef` command or the Deployment Environment Configuration wizard.

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgrAdmin -password dmgrPass
> $AdminTask addNodeToDeploymentEnvDef { -topologyName myDepEnv -nodeRuntime WESB
-topologyRole Messaging -nodeName MyNode}
```

Note: If you disable administrative security, you do not need to provide a user ID and password.

Generating deployment environments using the command line:

You can generate deployment environments using the **wsadmin** interface. This capability allows you to configure multiple deployment environments unattended on a deployment manager using a script.

Before you begin

You must enter the commands on the deployment manager on which you are configuring deployment environments.

Required security role for this task: When security and role-based authorization are enabled, you must log in to the administrative console as an administrator or configurator to perform this task.

About this task

After you have imported or created deployment environments on a deployment manager, you can configure the deployment environments using the **generateDeploymentEnv** command.

Procedure

1. Enter the **wsadmin** environment.
2. Enter the **generateDeploymentEnv** command for each topology you are configuring.

Example

The following command configures the `eastEnvironment` topology on host `myDmgr`.

```
wsadmin -connType SOAP -host myDmgr -port 8879
> $AdminTask generateDeploymentEnv {-topologyName eastTopology}
> $AdminConfig save
```

Note: If administrative security is enabled, you are prompted for a user ID and password after the system processes the **wsadmin** command.

What to do next

Save the configured deployment environments. From the command line, enter `$AdminConfig save`.

Validating the deployment environment definition from the command line:

You can validate the deployment environment definition using the **wsadmin** command.

Before you begin

The task assumes that the node has been federated to the deployment manager.

You must be on the deployment manager where you generated the deployment environment definition.

Required security role for this task: When security and role-based authorization are enabled, you must use a userid and password with administrator or operator authority to perform this task.

About this task

This task validates the deployment environment definition and uses the **wsadmin** command.

Procedure

1. Open a command window. The **wsadmin** command can be found in either of the following directories:
`install_root/profiles/dmgr profile/bin`
`install_root/bin`
2. At the command prompt, enter the **wsadmin** command to enter the **wsadmin** environment.
3. Enter the **validateDeploymentEnvDef** command to validate the deployment environment definition.

Note: If administrative security is on, you will be prompted for a user ID and password, if you do not supply it in the command.

Example

This example validates the deployment environment definition (**myDepEnv**) with administrative security enabled:

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgrAdmin -password -dmgrPass  
> $AdminTask validateDeploymentEnvDef { -topologyName topOne}
```

Note: If you disable administrative security, you do not need to provide a user ID and password.

Displaying deployment environment status using the command line:

You can display the current status of a deployment environment using the **wsadmin** command.

Before you begin

The admin client must connect to the deployment manager for which you are displaying the status.

Required security role for this task: When security and role-based authorization are enabled, you must use a userid and password with administrator or operator authority to perform this task.

About this task

This task displays the current status of a deployment environment and uses the **wsadmin** command.

Procedure

1. Open a command window. The **wsadmin** command can be found in either of the following directories:

```
install_root/profiles/dmgr profile/bin
install_root/bin
```

2. At the command prompt, enter the **wsadmin** command to enter the command environment.

Note: Make sure **wsadmin** connects to the correct deployment manager, when running in connected mode.

3. Use the **showDeploymentEnvStatus** command to show the current status of the deployment environment.

Note: If administrative security is on, you will be prompted for a user ID and password, if you do not supply it in the command.

The following table lists the results that might be returned.

Note: Some of the states listed in the table are valid for configured topologies only. The states that are apply to configured topologies only are noted as such.

Table 83. States of a topology instance in order of least to most available

| State | Description |
|----------------------|--|
| Incomplete | The deployment environment is not missing any elements but is incomplete in some way. Incomplete state may mean the deployment environment is missing a required role, node, comp or dependencies . The warning message contains additional details. |
| Complete | This state is also known as <i>Not configured</i> and it means that the configuration is known and complete but has not yet been generated. |
| Configured | This means the configuration is in synch. |
| Partially configured | The deployment environment has been generated but deferred configuration has not been completed. |
| Unknown | The system cannot determine the current state of the deployment environment. A resync operation could be performed on this state. |
| Stopped | State applies to configured topologies only.All deployment targets in the topology are stopped. |
| Running | State applies to configured topologies only.The deployment environment is available and all functions are running. |
| Partially started | State applies to configured topologies only.The deployment environment is available but at least one function is partially running. |
| Starting | State applies to configured topologies only.The deployment environment is starting. |
| Partially stopped | State applies to configured topologies only.The deployment environment is available but at least one function is stopped or partially stopped. |
| Stopping | State applies to configured topologies only.The deployment environment is stopping |
| Unavailable | State applies to configured topologies only.The deployment environment state is unavailable. |

Example

This example displays the status of a deployment environment (**MyDepEnv**) on the host (**myDmgr**) with administrative security enabled.

Note: If you are running the admin client from the deployment manager bin folder, you do not need to include the `-host` and `-port` parameters in the command.

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgradmin -password dmgrpass  
> $AdminTask showDeploymentEnvStatus {-topologyName myDepEnv}
```

The `-connType` parameter specifies the type of connection to be used; the default argument is SOAP. Because SOAP is the default, you do not need to give it explicitly.

The `-host` parameter specifies the host used for the SOAP or RMI connection. The default value for `-host` is the local host. If the node is running on the local host, you do not need to specify `-host`.

Note: If you disable administrative security, you do not need to provide a user ID and password.

Creating and configuring components

You can use the administrative console to create and configure components for WebSphere ESB.

Components can be configured in one of three ways:

1. Some WebSphere ESB components can be configured at profile creation time. This is particularly the case for a stand-alone server, for which most of the components may be configured at profile creation time.
2. For a network deployment cell, components may be configured by building and generating a Deployment Environment. By setting up a Deployment Environment, you build server clusters and configure the WebSphere ESB components on them.

Note: Deployment Environments are not available for a stand-alone server.

3. The WebSphere ESB servers and clusters (and the components that run on them) can be configured individually, using either the administrative console or administrative scripts.

This information in this section describes how to create components for WebSphere ESB manually using the administrative console.

Configuring SCA support for a server or cluster

Use the Service Component Architecture (SCA) console page to enable a server or cluster in a network deployment environment to host service applications, their required messaging engines and destinations, or both.

Before you begin

Before configuring SCA support, determine the following:

- Whether you are using a stand-alone server profile. If so, SCA support is already configured and you cannot use the Service Component Architecture page to remove that support; however, you can use this page to modify some properties for database data sources.

- Where to host the messaging engines and destinations (use either a local or remote bus member).
- Whether you need to configure the SCA system bus only, or whether you also need to configure the SCA application bus. The application bus is configured by default and is required if you plan to deploy SCA applications that use WebSphere Business Integration Adapters.

Security role required for this task: You must be logged in as administrator or configurator to perform the following task.

About this task

Service applications require the use of one or more of the automatically created service integration buses, which must have configured messaging engines for destinations. By default, new servers and clusters in a network deployment configuration are not configured to host SCA applications and their destinations.

To configure SCA support on your server or cluster, perform the following steps.

Procedure

1. From within the administrative console, click one of the following, depending on your scope:
 - **Servers > Server Types > WebSphere application servers > *server_name* > Service Component Architecture**
 - **Servers > Clusters > WebSphere application server clusters > *cluster_name* > Service Component Architecture**
2. Click **Support the Service Component Architecture components**.
3. In the Bus Member Location panel, specify where you want to host the destinations and messaging engines required by the SCA applications. There are two options:
 - **Local.** Specifies that you plan to host SCA applications, destinations, and messaging engines on the current server or cluster.
 - **Remote.** Specifies that you plan to host SCA applications on the current server or cluster while hosting destinations and messaging engines on a remote server or cluster (also referred to as a *deployment target*).
4. **(Remote bus member only)** If you selected **Remote** in the previous step, specify the remote server or cluster you want to use to host application destinations and messaging engines. Use the drop-down menu to select an existing deployment target (one that is already configured as a member of the SCA system bus), or click **New** to select a new server or cluster from the Browse Deployment Target page.

If you select a new server or cluster from the Browse Deployment Target page, the necessary messaging is automatically configured on that target when you complete the SCA configuration documented in this topic.

5. Use the table in the System Bus Member panel to verify or modify the system bus data source configuration.
 - a. Verify any default values in the **Database name**, **Schema**, **Create Tables**, **User name Password**, **Server**, and **Provider** fields. See the online help for detailed information about these fields and the values they accept.
 - b. If no default values exist in these fields, or if the default values are incorrect, enter the appropriate values for the system bus data source. You can enter values directly in the field or by clicking **Edit** and making edits on the Data Source details page.

- c. Optional: Ensure that the data source can contact and authenticate with the database by clicking **Test Connection**.
- 6. Use the table in the Application Bus Member panel to verify or modify the application bus data source configuration.
 - a. Ensure the **Enable the WebSphere Business Integration Adapter components** option is selected.

Note: If you do not want to use the application bus, clear the **Enable the WebSphere Business Integration Adapter components** option and proceed to Step 7.

- b. Verify any default values in the **Database name**, **Schema**, **Create Tables**, **User name Password**, **Server**, and **Provider** fields. See the online help for detailed information about these fields and the values they accept.
- c. If no default values exist in these fields, or if the default values are incorrect, enter the appropriate values for the application bus data source. You can enter values directly in the field or by clicking **Edit** and making edits on the Data Source details page.
- 7. Click **OK** to complete the SCA configuration.
- 8. Save your changes. You can also optionally review the changes you have made.

Considerations for Service Component Architecture (SCA) support in servers and clusters:

Servers and clusters can support Service Component Architecture (SCA) applications, application destinations, or both.

SCA applications (also called service applications) require the use of one or more of the automatically created service integration buses. Each application uses a set of messaging resources, which are called *destinations*. These destinations require configured messaging engines, and they can be hosted on the same server or cluster as the application or on a remote server or cluster. Messaging engines typically use database data sources; note that a file store can be used in place of a database data source in a stand-alone server profile if that option was selected during profile creation.

By default, new servers and clusters in a network deployment or managed node environment are not configured to host SCA applications and their destinations.

Note: A stand-alone server has SCA support automatically configured. You cannot disable this configuration.

To enable this support, use the Service Component Architecture page in the administrative console. For servers, ensure that the application class-loader policy is set to **Multiple**.

Before enabling SCA support for a server or cluster in a network deployment or managed node environment, determine which of the following possible configurations you want to implement:

- **Remote bus member configuration:** The server or cluster hosts SCA applications, but the destinations are hosted on a remote server or cluster. This scenario requires the remote service integration bus members to be configured with the messaging engines needed to host the destination.




While the use of remote messaging requires initial investment in planning for and configuring the service integration bus and its members, that configuration

can be reused by multiple members within the application cluster. Messages are distributed to every member. In addition, the initial configuration can be structured to provide failover support.

- **Local bus member configuration:** The server or cluster hosts both SCA applications and application destinations. The required messaging engines are configured using the local bus members on the server or cluster.

Refer to the planning topics to help you decide which configuration is appropriate for your environment.

Related information:

-  [Configuring class loaders of a server](#)
-  [Learning about service integration buses](#)
-  [Messaging engines](#)

Configuring Business Space

You can configure Business Space powered by WebSphere, which provides a common interface for application users to create, manage, and integrate web interfaces across the IBM Business Process Management portfolio, WebSphere Enterprise Service Bus, and other IBM products.

Configuring the relationship service

After installing the product, you need to set the configuration properties for the relationship service.

Before you begin

Required security role for this task: When security and role-based authorization are enabled, you must be logged in as a configurator or an administrator to perform this task. Any WebSphere security role can view this configuration.

About this task

To set the data source and query block size (relationship instance count) properties for the relationship service, perform the following steps.

Procedure

1. Ensure that the administrative console is running.
2. In the navigation pane, click **Integration Applications > Relationship Manager**.
3. Click **Relationship Services configuration**.
The configuration tabbed page displays, showing the name and version (read-only) of the currently installed relationship service.
4. In the **Query block size (relationship instance count)** field, specify the maximum cache that the relationship service should set aside for relationship queries. This setting determines the size of the query results set. By default, 5000 relationship instances are read at once. This field controls the server size memory usage and provides the administrator with a level of control over how much memory resource is consumable by any given query.
5. In the **Data source** field, specify the default data source for the relationship service by entering the Java Naming and Directory Interface (JNDI) name of a data source defined at the cell level. This is where the tables for the relationship service are stored. Each relationship-related schema is created in this data source by default.
6. You then have the following options:

- Click **OK** to save your changes and return to the previous page.
- Click **Reset** to clear your changes and restore the currently configured values or most recently saved values.
- Click **Cancel** to discard any unsaved changes on the page and return to the previous page.

Setting up the messaging server environment

Before running any XMS applications, including the sample applications provided with XMS, you must set up the messaging server environment.

About this task

The steps that you need to complete to set up the messaging server environment depend on the artifacts that an application connects to, and whether you are using the Message Service Client for .NET or the Message Service Client for C/C++. The steps are described in the documentation for the type of client.

Procedure

- Setting up for Message Service Client for .NET
- Setting up for Message Service Client for C/C++

What to do next

You can use the sample applications provided with the Message Service clients to verify your installation and messaging server setup. For more information about using the sample applications, see the following topics:

- Using .NET sample XMS applications
- Using C/C++ sample XMS applications

Configuring the JNDILookup web service

If you are using the administered JMS objects provided by WebSphere ESB with Message Service Clients for C/C++ and .NET, you must configure the JNDILookup web service that WebSphere ESB provides to enable non-Java clients to access administered JMS objects from a non-Java environment.

Before you begin

Before starting this task, make sure that the JNDILookup web service application has been installed.

About this task

Administratively-defined ConnectionFactory and Destination objects provide a separation between a JMS implementation and the JMS interfaces, which makes JMS client applications more portable since they are sheltered from the implementation details of a JMS provider. Administered objects enable an administrator to manage the connection settings for client applications from a central repository. For example, the specific queue that an application uses can be altered by changing the administered Destination object that the application obtains via JNDI.

Non-Java clients such as Message Service Clients for C/C++ and .NET can also use administered objects. However, since the administered JMS objects provided by WebSphere ESB are serialized Java objects accessed through JNDI, non-Java clients are not able to interpret them properly without the use of the JNDILookup web

service. This web service provides a lookup operation that allows Message Service Clients for C/C++ and .NET to request the retrieval of a JNDI object by specifying the name of the object. The properties of the administered object are returned to the application using a map of name/value pairs.

Procedure

Define the JNDILookup web service URL within the Message Service Client for C/C++ or Message Service Client for .NET application. To define the web service URL within an application, set the XMSC_IC_URL property of the InitialContext object to the web service endpoint URL. This property can alternatively be specified as an argument on constructing the InitialContext object.

Configuring a CEI database

You can configure a Common Event Infrastructure (CEI) database manually and use the CEI functionality for WebSphere ESB.

About this task

The procedure in this topic describes how to configure a CEI database for use with WebSphere ESB.

Procedure

1. After generating the database scripts, save your changes using \$AdminConfig save. In addition to generating the database scripts, the commands create JDBC resources for the CEI event service to use.
2. Copy the scripts that you generated to the database server. The directory location for the scripts depends on the scope where the CEI is deployed.
3. Log into the database server as a user with read and write access on the database. Open a command prompt and initialize the command line interface for the database software. To create the event database, run the script for your database type (for example cr_event_db2 server <db2_user>).

What to do next

Configuring WebSphere ESB widgets for WebSphere Portal

Your WebSphere ESB widgets can be displayed in WebSphere Portal.

Procedure

To display widgets in WebSphere Portal, complete the following high-level steps:

1. Configure Business Space.
2. Configure widgets to work with WebSphere Portal.

Configuring Common Event Infrastructure

You can configure Common Event Infrastructure resources, or change existing resources, using the server AdminTask object

About this task

Use the administrative console to configure CEI when you are installing it in a network deployment environment or in a cluster or in a stand-alone server configuration.

You can also use the `wsadmin` command to configure CEI, or you can use the command to alter an existing CEI configuration. In either case, you would change the configuration of CEI by using the server `AdminTask` object to run administrative commands.

After changing CEI configuration, you must restart the server or cluster.

Common Event Infrastructure components:

Common Event Infrastructure components are installed as a set of applications, services, and resources on the server.

When you configure Common Event Infrastructure, a number of components are created and deployed on your server.

Common Event Infrastructure service

A service installed into the server, that enables applications and clients to use Common Event Infrastructure. You can view the configuration of the Common Event Infrastructure service in the administrative console, as follows:

- For a server, click **Servers > Application Servers > *server_name* > Business Integration > Common Event Infrastructure > Common Event Infrastructure Service**.
- For a cluster, click **Servers > Clusters > *cluster_name* > Business Integration > Common Event Infrastructure > Common Event Infrastructure Service**.

If the check box labeled `Enable the event infrastructure server` is selected, then the service is installed and running or it will start after you restart your server or cluster. If it is cleared, then the service is not installed or will be uninstalled after you restart your server or cluster.

Event service settings

A set of properties used by the event service that enable event distribution and persistence using the data store. Typically, no configuration is necessary for this resource, but you might need to create additional event service settings if you want to set up multiple event services in the same cell. To view the event service settings, click **Service integration > Event service > Event service settings**.

Event messaging configuration

The resources that support asynchronous event transmission to the event service using the Java Messaging Service (JMS). The default messaging configuration uses the server embedded messaging. You can optionally configure an external JMS provider for event messaging.

Event database

The event database is used to persistently store events received by the event service. You can configure an external event database on the following products: DB2, Oracle, and SQLServer.

Event filter plug-in

A filter plug-in is used to filter events at the source using XPath event selectors. To configure the filter properties, click **Service Integration > Common Event Infrastructure > Event Emitter Factories > Event Filter Settings**.

Emitter factory

An emitter factory is an object used by event sources to create emitters; an

emitter is used to send events to the event service. The properties of an emitter factory affect the behavior of any emitter that is created using that emitter factory. To view the available emitter factories, click **Service Integration > Common Event Infrastructure > Event Emitter Factories**.

Event service transmission

An event service transmission is an object defining properties that determine how emitters access the event service synchronously using EJB calls; these properties are used by emitter factories when creating new emitters. You can view or change the available event service transmissions from the emitter factory settings.

JMS transmission

A JMS transmission is an object that defines properties that determine how emitters access the event service asynchronously using a JMS queue; these properties are used by emitter factories when creating new emitters. You can view or change the available JMS transmissions from the emitter factory settings.

Event group

An event group is a logical collection of events used to categorize events according to their content. When querying events from the event service or subscribing to event distribution, an event consumer can specify an event group to retrieve only the events in that group. Event groups can also be used to specify which events should be stored in the persistent data store. To view the available event groups in the administrative console, click **Service integration > Common Event Infrastructure > Event service > Event services > *event_service* > Event groups**.

Configuring the Common Event Infrastructure using the administrative console:

ConfigureCommon Event Infrastructure by using the server administrative console.

About this task

Open the Common Event Infrastructure Server panel of administrative console:

If you are configuring a server, select **Servers > Server Types > WebSphere application servers > *server_name* > Business Integration > Common Event Infrastructure > Common Event Infrastructure Server**.

If you are configuring a cluster, click **Servers > Clusters > WebSphere application server clusters > *cluster_name* > Business Integration > Common Event Infrastructure > Common Event Infrastructure Server**.

Procedure

1. Enable the deployment of the Common Event Infrastructure enterprise application by selecting the check box labeled **Enable the event infrastructure server**. If the server has already been configured, then you can enable or disable it by selecting or clearing the check box. If the enable check box is cleared then Common Event Infrastructure has not been configured, or has had a previous configuration disabled but the server has not been restarted. An information message shows you whether this deployment target has Common Event Infrastructure configured. If the server has already been configured, you can change the data source settings for the event database, the message store, or both.

Note: If you select the check box to enable the Common Event Infrastructure server and the server has not yet been configured, then the parameters shown is used to configure it unless you change them.

- If you are performing the configuration the first time, then the event data source tables are created on the common database. If there is already a Common Event Infrastructure server configuration, then you need to create a database.
- The messaging service is created under a unique schema under the common database.

When the server/cluster on which Common Event Infrastructure has been configured is restarted, then the new changes take effect.

2. Configure (or change the current settings for an existing configuration of) the event database by using one of the following methods to populate the fields with the appropriate settings.
 - Click **Edit** for a database configuration panel with a more extensive list of options than the ones listed on the panel.
 - Use the fields on the panel to enter the information:
 - a. **Database name** – the name of the database you use to store events.
 - b. **Create Tables** – select this check box if you want to create the database tables on the event database.

Note: If you are configuring Common Event Infrastructure to use a database on another server, then you are not be able to create the tables using this control. Instead, you will have to use the database scripts that will be generated after you complete the rest of this configuration. In this case, you can click **Edit** to show the data source detail panel, which tells you the location of the database creation scripts.

- c. **Username** and **Password** – for authenticating into the event database.
- d. **Server** – name of the server where the event database is located.
- e. **Provider** – choose a provider for your database from the menu.

Note: The **Schema** field is activated if the database is created using DB2 on the z/OS platform or on the iSeries platform. In all other cases, the schema field is disabled.

Important: If the tables exist on the target database, then the configuration can fail.

3. Select whether the Common Event Infrastructure bus is to be **Local** on the server, or **Remote** and reside on another server. If you choose remote, then select the remote location from the menu or click **New** to create a new remote bus.
4. Configure Common Event Infrastructure support for messaging.
 - Click **Edit** for a database configuration panel with a more extensive list of options than the ones listed on the panel.
 - Use the fields on the panel to enter the information:
 - a. **Database name** – enter the name of the database you use to store messages.
 - b. **Schema** – enter a name for the schema, or accept the default name given.
 - c. **Username** and **Password** – for authenticating into the messaging database.
 - d. **Server** – name of the server where the messaging database is located.
 - e. **Provider** – choose a provider for your database from the menu.

5. Create a messaging authentication alias for the Common Event Infrastructure bus.
 - a. Select **Additional Properties > JMS Authentication Alias**.
 - b. Enter the user ID and password you use for secure communications across the System Integration Bus. You can accept the default configured values of "CEI" for both the user ID and password if security is disabled. If security has been enabled, then enter the user ID and password used for the bus authentication. In a production environment, you would select your own user ID and password to secure the system.
 - c. Click **OK**.
6. Click **OK** or **Apply**.
7. Restart your server or cluster.

Results

All the major parts of Common Event Infrastructure are now configured and running on your server or cluster. The configuration includes the event data store, the messaging engine, and the event application. This single panel can be used in place of many commands and steps you would otherwise use to configure Common Event Infrastructure.

What to do next

After you have restarted your server or cluster, you will be able to store service component events that are emitted from your applications. You can now change the runtime properties of the Common Event Infrastructure server by selecting the **Common Event Infrastructure Destination** panel. You can choose whether to start the Common Event Infrastructure server at startup, and specify the emitter factory JNDI name where the events are sent.

Deploying the Common Event Infrastructure application:

Before you can use Common Event Infrastructure, you must first deploy the event service and associated resources in the server runtime environment.

About this task

The Common Event Infrastructure enterprise application includes the runtime components of the event service and the default messaging configuration used for asynchronous event submission.

To deploy the event service:

Procedure

From the wsadmin tool, run the **deployEventService** administrative command in batch or interactive mode. The parameters of the **deployEventService** administrative command are as follows:

nodeName

The name of the node where the event service to be deployed. This parameter is optional; if you do not specify a node name, the default is the current node. If you specify a node name, then you must also specify the server name using the **serverName** parameter. This parameter is not valid if you are deploying the event service in a cluster.

serverName

The name of the server where the event service to be deployed. This parameter is required only if you specify a node; it is not valid if you are deploying the event service in a cluster.

clusterName

The name of the cluster where the event service to be deployed. This parameter is optional and must not be specified if you are deploying at the node or server scope.

enable

Indicates whether the event service to be started automatically when the server starts. The default value is true.

Results

After the administrative command completes, the Common Event Infrastructure event service and default messaging configuration are deployed at the specified scope.

What to do next

If WebSphere security is enabled, you must also configure the JMS authentication alias and password using the **setEventServiceJmsAuthAlias** administrative command.

If you are deploying the event service in a cluster, you must also manually configure the event database.

Deploying Common Event Infrastructure in a cluster:

There are several ways you can deploy Common Event Infrastructure resources in a cluster environment.

Deploying Common Event Infrastructure in an existing cluster:

You can deploy the event service application in an existing cluster.

About this task

Deploying the event service application in a cluster is essentially the same as deploying the application on a stand-alone server. However, in a cluster environment, no default event database is configured.

To deploy and configure Common Event Infrastructure in a cluster environment:

Procedure

1. Run the **deployEventService** administrative command as you would for a stand-alone server, but specifying the name of the cluster. Use the **clusterName** parameter to specify the cluster.
2. On the deployment manager system, run the database configuration administrative command. Specify the cluster name using the **clusterName** parameter. This command generates the database configuration script.
3. Copy the generated database configuration script to the database system.
4. Run the database configuration script on the database system to create the event database.

5. On the deployment manager system, run the **enableEventService** command to enable the event service. Use the `clusterName` parameter to specify the name of the cluster.

Creating a cluster by converting an existing Common Event Infrastructure server:

You can create a cluster by converting an existing stand-alone server that is already configured with Common Event Infrastructure.

Before you begin

Before you can convert the existing server, make sure that it is fully configured for Common Event Infrastructure. The configuration includes deploying the event service application and configuring the event database.

About this task

To create the cluster:

Procedure

1. Follow the typical WebSphere process for converting a stand-alone server into the first member of a new cluster. When the server is converted, the following steps take place:
 - Common Event Infrastructure resources available at the scope of the server are moved to the new cluster scope.

Default database: If the existing server was configured with a non-supported database, the database resources are not moved to the cluster scope. Instead, these resources are removed. In this situation, the event service in the cluster is disabled by default.

 - The deployed event service application target list is modified to remove the converted server and add the new cluster.
2. Optional: If the converted server was configured with a non-supported database, you must configure a new event database for the cluster and then enable the event service:
 - a. On the deployment manager system, run the database configuration administrative command. Specify the cluster name using the `clusterName` parameter. This command generates the database configuration script.
 - b. Copy the generated database configuration script to the database system.
 - c. Run the database configuration script on the database system to create the event database.
 - d. On the deployment manager system, run the **enableEventService** command to enable the event service. Use the `clusterName` parameter to specify the name of the cluster.

Creating a cluster by using an existing Common Event Infrastructure server as a template:

You can create a cluster by specifying an existing Common Event Infrastructure server as a template.

Before you begin

Before you can create a cluster using this method, you must have an existing server that is fully configured for Common Event Infrastructure. The configuration includes deploying the event service application and configuring the event database.

About this task

To create the cluster:

Procedure

1. Follow the typical WebSphere process for creating new cluster, using the existing Common Event Infrastructure server as a template for the first cluster member. When the first member is created, the following steps take place:
 - Common Event Infrastructure resources available at the scope of the existing server are copied to the new cluster scope.
- Default database:** If the existing server was configured with a non-supported database, the database resources are not copied to the cluster scope. The default database configuration is not supported in a cluster. In this situation, the event service in the cluster is disabled by default.
- The deployed event service application target list is modified to include the new cluster.
2. Optional: If the existing server was configured with a non-supported database, you must configure a new event database for the cluster and then enable the event service:
 - a. On the deployment manager system, run the database configuration administrative command. Specify the cluster name using the `clusterName` parameter. This command generates the database configuration script.
 - b. Copy the generated database configuration script to the database system.
 - c. Run the database configuration script on the database system to create the event database.
 - d. On the deployment manager system, run the **enableEventService** command to enable the event service. Use the `clusterName` parameter to specify the name of the cluster.

Configuring event messaging:

You can modify the messaging configuration used for JMS transmission of events to the event service.

About this task

You will create the messaging infrastructure for Common Event Infrastructure when you use the administrative console panel to configure Common Event Infrastructure on a server. Generally, the messaging configuration will use the default messaging provider and create a single JMS queue for asynchronous transmission of events to the event service. You can, if necessary, modify this messaging configuration.

Configuring additional JMS queues:

If you are using the default event messaging configuration, you can add additional JMS queues for transmission of events to the event service.

About this task

To configure additional JMS queues to use the default messaging configuration, you can set up multiple JMS queues that are routed to the service integration bus queue destination. The Common Event Infrastructure service integration bus queue destination depends upon the scope at which the event service is deployed:

| Scope | Service integration bus queue destination |
|---------|--|
| Server | <code>node.server.CommonEventInfrastructureQueueDestination</code> |
| Cluster | <code>cluster.CommonEventInfrastructureQueueDestination</code> |

Configuring event messaging using an external JMS provider:

If you do not want to use the default embedded messaging configuration for event transmission, you can configure asynchronous message transport to use an external Java Messaging Service (JMS) provider.

Before you begin

Before you can configure event messaging using an external JMS provider, you must first create a JMS queue and connection factory using the appropriate interfaces for your JMS provider. You must also create a listener port or activation specification.

About this task

To configure event messaging using an external JMS provider:

Procedure

From the wsadmin tool, run the **deployEventServiceMdb** administrative command in batch or interactive mode. The parameters of the **deployEventServiceMdb** command are as follows:

applicationName

The application name of the event service message-driven bean to be deployed. This parameter is required.

nodeName

The name of the node where the event service message-driven bean is to be deployed. If you specify a node name, you must also specify a server name. The node name is an optional parameter; the default value is the current node. Do not specify this parameter if you are deploying the application in a cluster.

serverName

The name of the server where the event service message-driven bean is to be deployed. This parameter is required if you are deploying the application at server scope; otherwise it is optional. Do not specify a server name if you are deploying the application in a cluster.

clusterName

The name of the cluster where the event service message-driven bean is to be deployed. Specify this parameter only if you are deploying the application in a cluster.

listenerPort

The name of the listener port used by the event service message-driven bean to publish events. The specified listener port must exist. You must specify either a listener port or an activation specification, but not both.

activationSpec

The JNDI name of the activation specification used by the event service message-driven bean to publish events. The specified activation specification must exist. You must specify either a listener port or an activation specification, but not both.

qcfJndiName

The JNDI name of the JMS queue connection factory to be used by the event service message-driven bean. This parameter is required if you specify an activation specification; otherwise it is optional. If you specify a queue connection factory and a listener port, the queue connection factory must match the one configured for the listener port.

Results

The **deployEventServiceMdb** administrative command deploys the message-driven bean for the event service, configured for the specified listener port or activation specification. It also creates an emitter factory and JMS transmission using the external JMS configuration. Applications can use either the default emitter factory (which is configured to use the default messaging configuration) or the new emitter factory (which uses the external JMS provider).

What to do next

If you want to set up more than one JMS queue to the event service, you can run this command multiple times, specifying different enterprise application names and JMS queues. Each time you run the script, it deploys an additional message-driven bean and configures new resources to use the specified JMS queue.

Configuring the JMS authentication alias:

If WebSphere security is enabled and you want to use asynchronous JMS messaging to submit events to the event service, you must configure the JMS authentication alias.

About this task

To configure the JMS authentication alias:

Procedure

From the wsadmin tool, run the **setEventServiceJmsAuthAlias** administrative command in batch or interactive mode. The parameters of the **setEventServiceJmsAuthAlias** command are as follows:

userName

The name of the user to be used for the JMS authentication alias. This parameter is required.

password

The password of the user to be used for the JMS authentication alias. This parameter is required.

nodeName

The name of the node where you want to update or create the JMS authentication alias. If you specify a node name, you must also specify a server name. Do not specify a node name if you are configuring the authentication alias in a cluster.

serverName

The name of the server where you want to update or create the JMS authentication alias. This parameter is required only if you specify a node; it is not valid if you are configuring the authentication alias in a cluster.

clusterName

The name of the cluster where you want to update or create the JMS authentication alias. Specify this parameter only if you are configuring the authentication alias in a cluster; if you specify a cluster name, do not specify a node or server name.

Results

The JMS authentication alias used by the event service objects is updated at the specified scope; if the authentication does not exist, it is created using the specified values.

Configuring the event database:

You can configure the event data source using commands that are specific for each supported database product.

About this task

The event database is required to support persistence of events. If you did not use the Common Event Infrastructure configuration panel in the administrative console, you still have the option of creating the event database by using the commands described here.

Event database limitations:

Some limitations apply to configurations of the event database using certain database software.

Refer to the following table to see which limitations might apply to your environment.

Table 84. Event database limitations

| Database type | Limitations |
|---------------|--|
| Oracle | <ul style="list-style-type: none"> The Oracle 11 JDBC thin driver imposes some size restrictions for string values if you are using a Unicode character set. You may receive an Oracle ORA-01461 error when events containing large values (such as a long message attribute) are stored in the event database. For more information about this restriction, refer to the Oracle 11 documentation. To avoid this problem, use the Oracle 11 OCI driver or the Oracle 11 thin driver. Oracle database software treats a blank string as a NULL value. If you specify a blank string as an event attribute value, that string is converted to a NULL when it is stored in an Oracle event database. |
| SQL Server | <ul style="list-style-type: none"> The SQL Server database must be configured to use mixed authentication mode. Trusted connections are not supported. The XA stored procedures must be installed. These stored procedures are provided with the JDBC driver from Microsoft Corporation. The sqljdbc.dll file must be available in a directory specified on the PATH statement. This file is provided with the JDBC driver from Microsoft Corporation. The Distributed Transaction Coordinator (DTC) service must be started. |

Configuring a DB2 event database (Linux, UNIX, and Windows systems):

You can configure an external event database using DB2 Universal Database on a Linux, UNIX, or Windows system.

About this task

To configure a DB2 event database on a Linux, UNIX, or Windows system:

Procedure

1. Start the wsadmin tool.
2. Use the AdminTask object to run the **configEventServiceDB2DB** administrative command in batch or interactive mode. The minimum required parameters of the **configEventServiceDB2DB** command are as follows:

createDB

Indicates whether the administrative command creates and run the database configuration scripts. Specify true or false. If this parameter is set to false, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

nodeName

The name of the node that contains the server where the event service data source is created. If you specify a node name, you must also specify a server name. You must specify one of the following:

- Node name and server name
- Cluster name

serverName

The name of the server where the event service data source is created.

clusterName

The name of the cluster where the event service data source is created. If you specify a cluster name, do not specify node and server names.

jdbcClassPath

The path to the JDBC driver. Specify only the path to the driver file; do not specify the file name.

dbHostName

The host name of the server where the database is installed.

dbUser

The DB2 user ID to use when creating the event database. The specified user ID must have sufficient privileges to create and drop databases.

dbPassword

The DB2 password to use.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **configEventServiceDB2DB** administrative command.

Results

The administrative command creates the required data source at the specified scope; if you specified true for the createDB parameter, the command also runs the generated database configuration script to create the database.

The generated database configuration scripts are stored by default in the *profile_root/databases/event/node/server/dbscripts/db2* directory. (In a Network Deployment environment, these scripts are stored under the deployment manager profile directory.) If you specified a value for the optional outputScriptDir parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

Configuring a DB2 database on a z/OS system:

You can configure an event database on a z/OS system using DB2 database software.

Before you begin

To configure the DB2 database from a remote client, you must have the DB2 Connect product installed with the latest fix packs.

About this task

To configure the event database:

Procedure

1. Linux UNIX Windows If you are configuring the z/OS event database from a Linux, UNIX, or Windows client system, follow these steps to create and catalog the database:
 - a. On the z/OS system, use the DB2 administration menu to create a subsystem.

- b. Optional: Create the storage group you want to use for the event database. You can also use an existing storage group (for example, sysdefault).
 - c. Enable the 4 K, 8 K, and 16 K buffer pools you want to use for the event database.
 - d. Grant the necessary permissions to the user ID you want the data source to use. This user ID must have rights to access the database and storage group you created; it must also have permission to create new tables, table spaces, and indexes for the database.
 - e. Catalog the remote database. Run the following commands, either in a script or in a DB2 command-line window:


```
catalog tcpip node zosnode remote hostname server IP_port
      system db_subsystem
catalog database db_name as db_name at node zosnode authentication DCS
```

 For more information about how to catalog a node and its databases, refer to the DB2 Connect documentation.
 - f. Verify that you can establish a connection to the remote subsystem. You can run the following command to perform the verification:


```
db2 connect to subsystem user userid using password
```
 - g. Bind to the host database. Run the following commands:


```
db2 connect to db_name user userid using password
db2 bind db2_root/bnd/@ddcsmvs.lst blocking all sqlerror continue message
mvs.msg grant public
db2 connect reset
```

 For more information about binding a client to a host database, refer to the DB2 Connect documentation.
2. On the WebSphere system, start the wsadmin tool.
 3. Use the AdminTask object to run the **configEventServiceDB2ZOSDB** administrative command in batch or interactive mode. The minimum required parameters of the **configEventServiceDB2ZOSDB** command are as follows:

createDB

Linux **UNIX** **Windows** Indicates whether the administrative command creates and runs the database configuration scripts. This parameter applies only if you are running the administrative command from a Linux, UNIX, or Windows client system. Specify true or false.

If this parameter is set to false, or if you are running the command on the z/OS system, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

nodeName

The name of the node that contains the server where the event service data source is created. If you specify a node name, you must also specify a server name. You must specify one of the following:

- Node name and server name
- Cluster name

serverName

The name of the server where the event service data source is created.

clusterName

The name of the cluster where the event service data source is created. If you specify a cluster name, do not specify node and server names.

jdbcClassPath

The path to the JDBC driver. Specify only the path to the driver file; do not specify the file name.

dbHostName

The host name of the server where the database is installed.

dbUser

The DB2 user ID to use when creating the event database. The specified user ID must have sufficient privileges to create and drop databases.

dbPassword

The DB2 password to use.

dbPort

The DB2 instance port.

dbSubSystemName

The name of the database subsystem.

storageGroup

The storage group for the event database and the event catalog database.

eventDBName

The event database name to be created.

eventCatalogDBName

The event catalog database to be created.

bufferPool4K

The name of the 4K buffer pool.

bufferPool8K

The name of the 8K buffer pool.

bufferPool16K

The name of the 16K buffer pool.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **configEventServiceDB2ZOSDB** administrative command.

Results

The administrative command creates the required data source at the specified scope; if you are running the command on a Linux, UNIX, or Windows DB2 client and you specified true for the createDB parameter, the command also runs the generated database configuration script to create the database. On a z/OS system, you must use the SQL Processor Using File Input (SPUFI) facility to run the generated DDL files. The DDL files are stored in the *profile_root*/databases/event/*node/server/db2zos/ddl* directory.

The generated database configuration scripts are stored by default in the *profile_root*/databases/event/*node/server/dbscripts/db2zos* directory. (In a Network Deployment environment, these scripts are stored under the deployment manager profile directory.) If you specified a value for the optional outputScriptDir parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

What to do next

After you have finished configuring the database, you can use the server administrative console to test the database configuration. To perform this task, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Configuring a DB2 database on an iSeries system:

You can configure an event database on an iSeries system using DB2 database software.

About this task

If you are using a local iSeries server to configure a remote iSeries server, you must specify a remote database entry on your local server as an alias to the target database. To configure the event database:

Procedure

1. Start the wsadmin tool.
2. Use the AdminTask object to run the **configEventServiceDB2iSeriesDB** administrative command in batch or interactive mode. The minimum required parameters of the **configEventServiceDB2iSeriesDB** command are as follows:

createdB

Indicates whether the administrative command creates and run the database configuration scripts. Specify true or false. If this parameter is set to false, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

Limitation: The administrative command can automatically run the database configuration script only on the iSeries system. If you are running the command on a client system, an error is returned.

nodeName

The name of the node that contains the server where the event service data source is created. If you specify a node name, you must also specify a server name. You must specify one of the following:

- Node name and server name
- Cluster name

serverName

The name of the server where the event service data source is created.

clusterName

The name of the cluster where the event service data source is created. If you specify a cluster name, do not specify node and server names.

toolboxJdbcClassPath

The path to the IBM Toolbox for Java DB2 JDBC driver. Use this parameter only if you want to use the Toolbox for Java driver instead of the native JDBC driver. Specify only the path to the driver file; do not include the file name.

nativeJdbcClassPath

The path to the DB2 for iSeries native JDBC driver. Use this parameter only if you want to use the native JDBC driver instead of the Toolbox for Java driver. Specify only the path to the driver file; do not include the file name.

dbHostName

The host name of the server where the database is installed. This parameter is required if you are using the Toolbox for Java JDBC driver.

dbUser

The DB2 user ID to use when creating the event database. The specified user ID must have sufficient privileges to create and drop databases.

dbPassword

The DB2 password to use.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **configEventServiceDB2iSeriesDB** administrative command.

Results

The administrative command generates scripts to create the required database and data source at the specified scope. These scripts are stored by default in the *profile_root/databases/event/node/server/dbscripts/db2iseries* directory. If you specified a value for the optional *outputScriptDir* parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

What to do next

If you ran the database configuration administrative command on a client system, you must transfer the generated scripts to the iSeries system and run them to create the required resources.

After you have finished configuring the database, you can use the server administrative console to test the database configuration. To do test the configuration, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Configuring an Oracle event database:

You can configure an external event database using Oracle Database on a Linux, UNIX, or Windows system.

Before you begin

Before you configure an Oracle event database, you must first create the database. The Oracle SID must already exist before you run the event database configuration command. The default SID for the event database is *event*.

About this task

To configure an Oracle event database:

Procedure

1. Start the wsadmin tool.
2. Use the AdminTask object to run the **configEventServiceOracleDB** administrative command in batch or interactive mode. The minimum required parameters of the **configEventServiceOracleDB** command are as follows:

createDB

Indicates whether the administrative command should create and run the

database configuration scripts. Specify `true` or `false`. If this parameter is set to `false`, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

nodeName

The name of the node that contains the server where the event service data source should be created. If you specify a node name, you must also specify a server name. You must specify one of the following:

- Node name and server name
- Cluster name

serverName

The name of the server where the event service data source should be created.

clusterName

The name of the cluster where the event service data source should be created. If you specify a cluster name, do not specify node and server names.

jdbcClassPath

The path to the JDBC driver. Specify only the path to the driver file; do not specify the file name.

oracleHome

The ORACLE_HOME directory. This parameter is required only if you specified `true` for the `createDB` parameter.

dbPassword

The password to use for the schema user ID created during the database configuration (the default user ID is `ceiuser`). This password is used to authenticate the Oracle database connection.

sysUser

The Oracle SYSUSER user ID. This user ID must have SYSDBA privileges.

sysPassword

The password for the specified SYSUSER user ID.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **configEventServiceOracleDB** administrative command.

Results

The administrative command creates the required data source at the specified scope; if you specified `true` for the `createDB` parameter, the command also runs the generated database configuration script to create the database.

The generated database configuration scripts are stored by default in the `profile_root/databases/event/node/server/dbscripts/oracle` directory. (In a Network Deployment environment, these scripts are stored under the deployment manager profile directory.) If you specified a value for the optional `outputScriptDir` parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

Configuring a SQL Server event database:

You can configure an external event database using Microsoft SQL Server Enterprise on a Windows system.

About this task

To configure a SQL Server event database:

Procedure

1. On the SQL Server database server system, create the directory used to contain the database files. By default, the files are written to the c:\program files\ibm\event\ceiinst1\sqlserver_data directory. If you need to specify a different location, you must edit the generated database configuration script to modify the value of the `ceiInstancePrefix` parameter, and then run the script manually.
2. On the server system, start the `wsadmin` tool.
3. Use the `AdminTask` object to run the **`configEventServiceSQLServerDB`** administrative command in batch or interactive mode. The minimum required parameters of the **`configEventServiceSQLServerDB`** command are as follows:

`createDB`

Indicates whether the administrative command should create and run the database configuration scripts. Specify `true` or `false`. If this parameter is set to `false`, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

`nodeName`

The name of the node that contains the server where the event service data source should be created. If you specify a node name, you must also specify a server name. You must specify one of the following:

- Node name and server name
- Cluster name

`serverName`

The name of the server where the event service data source should be created. If you specify a server name, you must also specify a node name.

`clusterName`

The name of the cluster where the event service data source should be created. If you specify a cluster name, do not specify node and server names.

`dbServerName`

The server name of the SQL Server database. This parameter is required only if you specified `true` for the `createDB` parameter.

`dbHostName`

The host name of the server where the SQL Server database is running.

`dbPassword`

The password to use for the user ID created to own the event database tables (the default user ID is `ceiuser`). The WebSphere data source uses this password to authenticate the SQL Server database connection.

`saUser`

A user ID with privileges to create and drop databases and users. This parameter is required only if you specified `true` for the `createDB` parameter.

`saPassword`

The password for the specified SA user.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **`configEventServiceSQLServerDB`** administrative command.

Results

The administrative command creates the required data source at the specified scope; if you specified `true` for the `createDB` parameter, the command also runs the generated database configuration script to create the database.

The generated database configuration scripts are stored by default in the `profile_root/databases/event/node/server/dbscripts/dbscripts/sqlserver` directory. (In a Network Deployment environment, these scripts are stored under the deployment manager profile directory.) If you specified a value for the optional `outputScriptDir` parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

Manually running database configuration scripts:

You can manually run the scripts generated by the database configuration administrative commands at any time.

About this task

Database configuration is a two-step process. The database configuration administrative command first generates a database-specific script for your environment; this generated script then configures the event database and data sources. If you specify `true` for the `createDB` parameter when running the administrative command, both steps happen automatically.

However, if you specify `false` for the `createDB` parameter, you must complete the database configuration by manually running the generated script on the target system. You might need to run the script manually in any of the following situations:

- You need to configure the event database on a different system from the one where you ran the administrative command.
- You need to re-create the event database at a later time.
- You need to modify the default options used by the generated script before running it.

Manually creating a DB2 event database on a Linux, UNIX, or Windows system:

Use the **`cr_event_db2`** to manually generate a database configuration script for a DB2 event database on a Linux, UNIX, or Windows server.

About this task

To manually run the generated database configuration script for a DB2 event database on a Linux, UNIX, or Windows system:

Procedure

1. On the server system, go to the directory containing the generated script. The default location is the `install_root/profiles/profile_name/dbscripts/CEI_ceiDbName` directory; if you specified a value for the `outputScriptDir` parameter of the database configuration administrative command, the scripts are stored in that location instead.
2. Using an ASCII text editor, make any required modifications to the configuration script. The name of the script varies depending upon the operating system in use:

- **Windows** Windows systems: `cr_event_db2.bat`
 - **Linux** **UNIX** Linux and UNIX systems: `cr_event_db2.sh`
3. Run the database creation script using the following syntax (remember to specify the file extension, if applicable):
`cr_event_db2 [client|server] db_user [db_password]`

The parameters are as follows:

client|server

Indicates whether the database is a client or server. You must specify either **client** or **server**.

db_user

The database user ID. This parameter is required.

db_password

The password for the database user. If you do not specify a password for a client database, you are prompted for it.

For example, the following command would create the DB2 event database for a client database, using the user ID `db2admin` and the password `mypassword`:

```
cr_event_db2 client db2admin mypassword
```

4. Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

What to do next

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Manually creating a DB2 event database on a z/OS system:

Use the **cr_event_db2zos** to manually generate a database configuration script for a DB2 event database on a z/OS system, using a Linux, UNIX, or Windows client system.

Procedure

To manually run the generated database configuration script for a DB2 event database on a z/OS system, using a Linux, UNIX, or Windows client system:

1. On the server system, go to the directory containing the generated script. The default location is the `install_root/profiles/profile_name/dbscripts/CEI_ceiDbName` directory. If you specified a value for the `outputScriptDir` parameter of the database configuration administrative command, the scripts are stored in that location instead.
2. Using an ASCII text editor, make any required modifications to the configuration script. The name of the script varies depending upon the operating system in use:
 - Windows systems: `cr_event_db2zos.bat`
 - Linux and UNIX systems: `cr_event_db2zos.sh`
3. Run the database creation script using the following syntax (remember to specify the file extension, if applicable):
`cr_event_db2zos [dbName=db_name] db_user [db_password]`

The parameters are as follows:

db_name

The database name to use. This parameter is optional; if you do not specify a database name, a name is generated.

db_user

The database user ID to use. This parameter is required.

db_password

The password for the database user. If you do not specify the password, the DB2 database prompts you for it.

For example, the following command would create a DB2 event database called event, using the user ID db2admin and the password mypassword:

```
cr_event_db2zos dbName=client db2admin mypassword
```

- Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

What to do next

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Manually creating a DB2 event database on an iSeries system:

Use the **cr_event_db2iseries** command to manually generate a database configuration script for a DB2 event database on an iSeries system

About this task

To manually run the generated database configuration script for a DB2 event database on an iSeries system:

Procedure

- On the server system, go to the directory containing the generated script. The default location is the *install_root/profiles/profile_name/dbscripts/CEI_ceiDbName* directory. If you specified a value for the outputScriptDir parameter of the database configuration administrative command, the scripts are stored in that location instead.
- Using an ASCII text editor, make any required modifications to the cr_event_db2iseries script.
- Start the Qshell interpreter.
- Run the database creation script using the following syntax:

```
cr_event_db2iseries db_user db_password
```

The parameters are as follows:

db_user

The database user ID. This parameter is required.

db_password

The password for the database user. This parameter is required.

For example, the following command would create the DB2 event database using the user ID db2admin and the password mypassword:

```
cr_event_db2iseries db2admin mypassword
```

5. Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

What to do next

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Manually creating an Oracle event database:

Use the **cr_event_oracle** command to manually generate a database configuration script for an Oracle event database.

About this task

To manually run the generated database configuration script for an Oracle event database:

Procedure

1. On the server system, go to the directory containing the generated script. The default location is the *install_root/profiles/profile_name/dbscripts/CEI_ceiDbName* directory. If you specified a value for the `outputScriptDir` parameter of the database configuration administrative command, the scripts are stored in that location instead.
2. Using an ASCII text editor, make any required modifications to the configuration script. The name of the script varies depending upon the operating system in use:
 - **Windows** Windows systems: `cr_event_oracle.bat`
 - **Linux** **UNIX** Linux and UNIX systems: `cr_event_oracle.sh`
3. Run the database creation script using the following syntax (remember to specify the file extension, if applicable):

```
cr_event_oracle password sys_user
                  sys_password [sid=sid]
                  [oracleHome=oracle_home]
```

The parameters are as follows:

password

The password for the schema user ID. This parameter is required.

sys_user

The user ID that has SYSDBA privileges in the Oracle database (typically the sys user). This parameter is required.

sys_password

The password for the specified sys user ID. If this user ID does not use a password, type none.

sid=sid

The Oracle system identifier (SID). This parameter is optional.

oracleHome=oracle_home

The Oracle home directory. This parameter is optional; if you do not specify a value, a generated path is used.

For example, the following command would create the Oracle event database using the schema user ID `auser` and the sys user ID `sys`:

```
cr_event_oracle auser sys syspassword sid=event oracleHome=c:\oracle
```

4. Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

What to do next

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Manually creating a SQL Server event database:

Use the **cr_event_mssql** command to manually generate a database configuration script for a SQL Server event database.

About this task

To manually run the generated database configuration script for a SQL Server event database:

Procedure

1. On the server system, go to the directory containing the generated script. The default location is the `install_root/profiles/profile_name/dbscripts/CEI_ceiDbName` directory. If you specified a value for the `outputScriptDir` parameter of the database configuration administrative command, the scripts are stored in that location instead.
2. Using an ASCII text editor, make any required modifications to the `cr_event_mssql.bat` script.
3. Run the database creation script using the following syntax:

```
cr_event_mssql user_id password [server=server] sauser=sa_user  
sapassword=sa_password
```

The parameters are as follows:

user_id

The SQL Server login user ID that will own the created tables. This user ID must be created in SQL Server so that a JDBC connection can be made to the database. (The JDBC drivers do not support trusted connections.)

password

The password for the new login user ID that is created.

server=server

The name of the server that contains the SQL Server database. This parameter is optional; the default value is the local host.

sauser=sa_user

The sa user ID. This user ID must have sufficient privileges to create databases and user logins.

sapassword=sa_password

The sa password, if using mixed authentication mode. If the sa user ID does not have a password set, specify `sapassword=` with no value. Omit this parameter if you are using a trusted connection.

For example, the following command would create the SQL Server event database using the login user ID `userid`:

```
cr_event_mssql userid apassword server=myserver sauser=sa sapassword=sapassword
```

4. Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

What to do next

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Upgrading the event database from a previous version:

If you have migrated from a previous version of Common Event Infrastructure and you are using event persistence, you might need to upgrade an existing event database.

About this task

Upgrading the event database is required if you are migrating from Common Event Infrastructure version 5.1 or earlier.

The database upgrade process upgrades the schema and metadata of the existing event database to the current version while preserving existing event data.

The database upgrade script upgrades the schema and metadata of the existing event database to the current version.

Unsupported versions: If your event database uses a version of database software that is no longer supported by Common Event Infrastructure 6.0, you must first migrate the database to a supported version using the appropriate procedure for the database software. You can then follow the event database upgrade process to upgrade the database.

Upgrading a DB2 event database from a previous version:

If you have an existing DB2 event database from Version 5.1 of Common Event Infrastructure on a Linux, UNIX, or Windows system, you must upgrade it to the current version.

About this task

To upgrade a DB2 event database on a Linux or UNIX system:

Procedure

1. Make a backup copy of the existing event database.
2. Go to the `profile_root/bin` directory.
3. Run the DB2 upgrade script for your operating system:
 - **Windows** Windows systems:

```
eventUpgradeDB2 runUpgrade=[true|false] dbUser=user
                  [dbName=name] [dbPassword=pw]
                  [dbNode=node] [scriptDir=dir]
```
 - **Linux** **UNIX** Linux and UNIX systems:

```
eventUpgradeDB2.sh runUpgrade=[true|false] dbUser=user  
[dbName=name] [dbPassword=pw]  
[dbNode=node] [scriptDir=dir]
```

The typical required parameters are as follows:

runUpgrade

Indicates whether you want the upgrade script to automatically run the generated DDL scripts to complete the database upgrade. This parameter is required. Specify false if you want to manually perform the database upgrade at a later time or on a different system.

dbUser

Specifies the DB2 user ID to use. This parameter is required.

dbName

Specifies the DB2 database name. The default name for the event database is event. This parameter is required if you specified runUpgrade=true.

dbPassword

Specifies the password for the specified DB2 user ID. This parameter is optional; if you do not specify a password, DB2 prompts you to type it.

dbNode

Specifies the database node name. This parameter is required if you are running the upgrade script from a DB2 client system.

scriptDir

Specifies the directory you want to contain the generated DDL scripts. This parameter is optional; if you do not specify a directory, the scripts are stored in the .\eventDBUpgrade\db2 directory.

To see a complete list of parameters and usage information, run the **eventUpgradeDB2** script with no parameters.

Results

The upgrade script generates the required DDL scripts for upgrading the event database. If you specified runUpgrade=true, the DDL scripts are automatically run, completing the upgrade.

Example

The following example upgrades an existing DB2 database on a Windows system:
eventUpgradeDB2 runUpgrade=true dbUser=db2inst1 dbName=event

What to do next

If you specified runUpgrade=false, you must manually run the DDL scripts on the database system to complete the database upgrade.

Upgrading a DB2 for z/OS event database from a previous version:

If you have an existing DB2 event database from Version 5.1 of Common Event Infrastructure on a z/OS system, you must upgrade it to the current version.

About this task

To upgrade a DB2 event database on a z/OS system:

Procedure

1. Make a backup copy of the existing event database.
2. Go to the *profile_root/bin* directory.
3. Run the DB2 for z/OS upgrade script for your client operating system:

- **Windows** Windows systems:

```
eventUpgradeDB2ZOS runUpgrade=[true|false] dbUser=user  
[dbName=name] [dbPassword=pw]  
[scriptDir=dir] storageGroup=group  
bufferPool4K=4kbufpool bufferPool8k=8kbufpool  
bufferPool16K=16kbufpool
```

- **Linux** **UNIX** Linux and UNIX systems:

```
eventUpgradeDB2ZOS.sh runUpgrade=[true|false] dbUser=user  
[dbName=name] [dbPassword=pw]  
[scriptDir=dir] storageGroup=group  
bufferPool4K=4kbufpool bufferPool8k=8kbufpool  
bufferPool16K=16kbufpool
```

The typical required parameters are as follows:

runUpgrade

Indicates whether you want the upgrade script to automatically run the generated DDL scripts to complete the database upgrade. This parameter is required. Specify false if you want to manually upgrade the database at a later time or on a different system.

z/OS systems: This parameter is ignored on a native z/OS system. Automatically running the generated DDL scripts is supported only on a client system.

dbUser

Specifies the DB2 user ID to use. This parameter is required.

dbName

Specifies the DB2 database name. The default name for the event database is event. This parameter is required if you specified runUpgrade=true.

dbPassword

Specifies the password for the specified DB2 user ID. This parameter is optional; if you do not specify a password, DB2 prompts you to type it.

scriptDir

Specifies the directory you want to contain the generated DDL scripts. This parameter is optional; if you do not specify a directory, the scripts are stored in the *.\eventDBUpgrade\db2zos* directory.

storageGroup

Specifies the name of the storage group. This parameter is required.

bufferPool4K

Specifies the name of the 4K buffer pool. This parameter is required.

bufferPool8K

Specifies the name of the 8K buffer pool. This parameter is required.

bufferPool16K

Specifies the name of the 16K buffer pool. This parameter is required.

To see a complete list of parameters and usage information, run the **eventUpgradeDB2ZOS** script with no parameters.

Results

The upgrade script generates the required DDL scripts for upgrading the event database. If you specified `runUpgrade=true` on a client system, the DDL scripts are automatically run, completing the upgrade.

Example

The following example upgrades a DB2 for z/OS event database from a Windows client system:

```
eventUpgradeDB2ZOS runUpgrade=true dbUser=db2inst1 dbName=event  
storageGroup=sysdeflt bufferPool4K=BP9 bufferPool8K=BP8K9 bufferPool16K=BP16K9
```

What to do next

If you specified `runUpgrade=false`, or if you ran the upgrade script on the z/OS system, you must manually run the generated DDL scripts on the z/OS system using the SQL Processor Using File Input (SPUFI) facility. This step completes the database upgrade.

Upgrading an Oracle event database from Version 5:

If you have an existing Oracle event database from Version 5.1 of Common Event Infrastructure, you must upgrade it to the current version.

About this task

To upgrade an Oracle event database:

Procedure

1. Make a backup copy of the existing event database.
2. Go to the *profile_root*/bin directory.
3. Run the Oracle upgrade script for your operating system:

- Windows systems:

```
eventUpgradeOracle runUpgrade=[true|false] schemaUser=schemauser  
[oracleHome=dir] [dbName=name]  
[dbUser=sysuser] [dbPassword=pw]  
[scriptDir=dir]
```

- Linux and UNIX systems:

```
eventUpgradeOracle.sh runUpgrade=[true|false] schemaUser=schemauser  
[oracleHome=dir] [dbName=name]  
[dbUser=sysuser] [dbPassword=pw]  
[scriptDir=dir]
```

The typical required parameters are as follows:

runUpgrade

Indicates whether you want the upgrade script to automatically run the generated DDL scripts to complete the database upgrade. This parameter is required. Specify `false` if you want to manually upgrade the database at a later time or on a different system.

schemaUser

Specifies the Oracle user ID that owns the database tables. This parameter is required.

oracleHome

Specifies the Oracle home directory. This parameter is required if you specified `runUpgrade=true`.

dbName

Specifies the Oracle database name. The default name for the event database is `event`. This parameter is required if you specified `runUpgrade=true`.

dbUser

Specifies the Oracle sys user ID. This parameter is required if you specified `runUpgrade=true`.

dbPassword

Specifies the password for the sys user ID. Do not specify this parameter if the sys user ID has no password.

scriptDir

Specifies the directory you want to contain the generated DDL scripts. This parameter is optional; if you do not specify a directory, the scripts are stored in the `.\eventDBUpgrade\oracle` directory.

To see a complete list of parameters and usage information, run the **eventUpgradeOracle** script with no parameters.

Results

The upgrade script generates the required DDL scripts for upgrading the event database. If you specified `runUpgrade=true`, the DDL scripts are automatically run, completing the upgrade.

Example

The following example upgrades an existing Oracle database on a Windows system:

```
eventUpgradeOracle runUpgrade=true schemaUser=cei  
dbName=event dbUser=sys
```

What to do next

If you specified `runUpgrade=false`, you must manually run the DDL scripts on the database system to complete the database upgrade.

Configuring WebSphere Business Integration Adapters

You must perform installation and configuration procedures for the WebSphere Business Integration Adapter to work with WebSphere ESB.

Procedure

1. Install the adapter.
 - a. Follow the procedures outlined at Installing WebSphere Business Integration Adapters products, which describe how to install WebSphere Business Integration Adapters.
 - b. Determine whether there are any additional required procedures that are specific to your adapter by going to the WebSphere Business Integration Adapters documentation and expanding the navigation under **Adapters**. If any additional installation tasks are listed for your adapter, perform those tasks.

2. Configure your adapter by going to the WebSphere Business Integration Adapters documentation, expanding the navigation under **Adapters**, and following the configuration instructions for your adapter. The configuration procedure generates the required artifacts.
3. Install the application EAR file by following the instructions for Deploying a mediation module.

Setting up administration of WebSphere Business Integration Adapters:

You must perform several administrative functions before you can manage a WebSphere Business Integration Adapter.

Before you begin

- You must be familiar with the procedures outlined in Installing WebSphere Business Integration Adapters products.
- You must have installed the application EAR file to create the artifacts required for the WebSphere Business Integration Adapter before you perform this task.

About this task

In order to have administrative control over a WebSphere Business Integration Adapter, perform the administrative functions in the Procedure section.

Procedure

1. Create a Queue Connection Factory.

From the top level of the administrative console, follow these steps:

- a. Expand **Resources**.
- b. Expand **JMS**.
- c. Select **Queue connection factories**.
- d. Select the scope level that matches the scope level of the Administration Input/Output Queues.
- e. Click **New** to create a new JMS queue connection factory.
- f. Choose the JMS resource provider. Select **Default messaging provider**, and click **OK**.
- g. Accept all the default values with the following exceptions:
 - **Name:** QueueCF
 - **JNDI Name:** jms/QueueCF
 - **BusName:** *Your bus name*
- h. Complete the creation of your new JMS queue connection factory by clicking **OK**.

A message window appears at the top of the JMS queue connection factory panel.
- i. Apply the changes that you have made at the local configuration level to the master configuration by clicking **Save** in the message window.

2. Create a WebSphere Business Integration Adapter resource.

From the top level of the administrative console, follow these steps:

- a. Expand **Resources**.
- b. Open the WebSphere Business Integration Adapters page.

Select **WebSphere Business Integration Adapters**.
- c. Create a new WebSphere Business Integration Adapter by clicking **New**.

- d. Accept all the default values with the following exceptions:
 - **Name:** EISConnector
 - **Queue connection factory JNDI name:** jms/QueueCF
 - **Administration input queue JNDI name:** *connectorName/AdminInQueue*
 - **Administration output queue JNDI name:** *connectorName/AdminOutQueue*
 - e. Complete the creation of the WebSphere Business Integration Adapter by clicking **OK**.

A message window appears at the top of the WebSphere Business Integration Adapters panel.
 - f. Apply the changes that you have made at the local configuration level to the master configuration by clicking **Save** in the message window.
3. Enable the WebSphere Business Integration Adapter Service.

From the top level of the administrative console, follow these steps:

 - a. Expand **Servers**.
 - b. Expand **Server types**.
 - c. Select **WebSphere application servers**.
 - d. From the list of servers, select a server where the WebSphere Business Integration Adapter Service is to be enabled.

Click the name of the server that hosts the resources of interest.
 - e. From the **Business Integration** list on the Configuration tab, select **WebSphere Business Integration Adapter Service**.
 - f. Ensure that the **Enable service at server startup** check box is selected.
 - g. Click **OK**.

A message window appears at the top of the WebSphere Business Integration Adapters page.
 - h. Repeat steps 3d to 3g for each server on which the WebSphere Business Integration Adapter Service is to be enabled.
 - i. Apply the changes that you have made at the local configuration level to the master configuration by clicking **Save** in the message window.

Note: When you enable or disable a WebSphere Business Integration Adapter service, you must restart the server in order for the changes to take effect.

Configuring WebSphere ESB for Service Federation Management

You can enable WebSphere ESB as a connectivity server that can be administered by the Service Federation Management (SFM) console provided with WebSphere Service Registry and Repository version 7.0. The SFM console can then configure SFM proxies in WebSphere ESB.

About this task

You might have separate enterprise service buses (ESBs) in different business units. Each ESB and associated service registry constitute a separate domain of connected service applications. This can result in expensive duplication of applications between domains and also in increased development effort to implement application connectivity across domains. SFM, provided in WebSphere Service Registry and Repository version 7.0, allows you to establish bridges between separate ESBs, allowing services and applications to be shared between domains.

SFM provides:

- A federation model which provides a unifying view of federation relevant content.
- A Service Connectivity Management Protocol, which accesses the service connectivity and registry components supporting a domain.
- A console for controlling service domains.

SFM allows the console user to configure services in one domain so that they are available to service consumers in another domain; the service endpoints in one domain are available as service proxy endpoints in another domain.

Configuring the Service Connectivity Management connectivity server:

The Service Federation Management (SFM) console uses the Service Connectivity Management Protocol (SCMP) to communicate with WebSphere ESB.

About this task

WebSphere ESB exposes the Atom based protocol as a system REST service named SCM Connectivity Server. This service is enabled by default in the REST service provider for stand-alone servers and the deployment manager of a Network Deployment environment.

Procedure

1. Configure the REST services. The Atom documents returned by the protocol contain absolute URLs which are retained by the SFM console. The protocol, host name, and port number used in those absolute URLs are taken from the REST service configuration. It is important to consider any load balancing and network components between the SFM console server and WebSphere ESB.
 - a. Configure the protocol, fully-qualified host name, and port number, for the stand-alone server or deployment manager REST service provider as described in the Configuring REST services in a service provider topic.
2. Provide the SFM console user with details to access the connectivity server.
 - a. The URL of the Atom service document for the connectivity server can be found on the REST services panel. The service has the type *SCM Connectivity Server*.
 - b. If WebSphere ESB administrative security is enabled, the SFM console user will also require a username and password to access the service endpoint. These credentials must be for a user in the RestServicesUser group who has sufficient administrative rights to install Service Connectivity Architecture modules.

Configuring the Service Connectivity Management connectivity provider:

You can configure all Service Connectivity Management (SCM) connectivity providers for your environment by using the administrative console.

About this task

An SCM connectivity provider is a logical partition of the ESB that is exposed via the SCM Protocol. It defines the target (server or cluster) to which proxy gateway modules will be deployed when a SCM group proxy is created on that connectivity provider. It also defines properties that will be used for proxy targets created on those group proxies.

Procedure

Select **Service integration > SCM connectivity providers**. The SCM connectivity providers page opens, displaying all connectivity providers in your environment.

Results

SCM connectivity providers can be added, removed, or worked with from this page.

Adding a connectivity provider:

You can add a server or a cluster as a Service Connectivity Management (SCM) connectivity provider using the administrative console.

Procedure

1. Click **Service integration > SCM connectivity providers**. The SCM connectivity providers page opens, displaying all connectivity providers in your environment.
2. Click **Add** to add a server or a cluster as a connectivity provider. The wizard for adding connectivity providers will open.
3. Complete **Step 1. Select a server or cluster** on the wizard to identify the server or cluster to which SCM group proxies for this connectivity provider should be deployed. Click **Next**.
4. Complete **Step 2. Specify SCM connectivity provider properties** on the wizard to specify the properties:

| Option | Description |
|---------------------|---|
| Name | The name of the SCM connectivity provider. This must be unique within the cell. An exception is thrown if the name already exists. The name, description, contact, organization and location will be visible to users of the Service Federation Management console. |
| Description | A brief description of the SCM connectivity provider. This is optional and defaults to an empty string. The name, description, contact, organization and location will be visible to users of the Service Federation Management console. |
| Contact | The name of a contact person for the SCM connectivity provider. This is optional and defaults to an empty string. The name, description, contact, organization and location will be visible to users of the Service Federation Management console. |
| Organization | The name of the owning organization for the SCM connectivity provider. This is optional and defaults to an empty string. The name, description, contact, organization and location will be visible to users of the Service Federation Management console. |

| Option | Description |
|-----------------------------|---|
| Location | The location for the SCM connectivity provider. This is optional and defaults to an empty string. The name, description, contact, organization and location will be visible to users of the Service Federation Management console. |
| HTTP host | The host name that will be returned for the endpoint of an insecure proxy target. This should be the host that web service clients in another domain will use to access the proxy, taking in to account web servers and other network components. |
| HTTP port | The port that will be returned for the endpoint of an insecure proxy target. This should be the port that web service clients in another domain will use to access the proxy, taking in to account web servers and other network components. |
| HTTPS host | The host name that will be returned for the endpoint of a secure proxy target. This should be the host that web service clients in another domain will use to access the proxy, taking in to account web servers and other network components. |
| HTTPS port | The port that will be returned for the endpoint of a secure proxy target. This should be the port that web service clients in another domain will use to access the proxy, taking in to account web servers and other network components. |
| Authentication Alias | The name of the authentication alias that will provide the basic authentication credentials used to retrieve WSDL documents via HTTP from the service registry associated with the SCM connectivity provider's domain. This parameter need not be specified if basic authentication is not required to connect to the service registry. |
| SSL configuration | The name of the SSL configuration used to retrieve WSDL documents via HTTP from a secure service registry associated with the SCM connectivity provider's domain. This is optional and, if not specified, the server's default SSL configuration will be used. |

5. Click **Finish**. The SCM connectivity provider page will open, with the new connectivity provider listed.
6. Review the **Messages** section to ensure that the connectivity provider and its properties are complete.
7. Click **Save** to save the connectivity provider to the master configuration.

Removing a connectivity provider:

You can remove a server or a cluster as a Service Connectivity Management (SCM) connectivity provider using the administrative console.

Procedure

1. Click **Service integration > SCM connectivity providers**. The SCM connectivity providers page opens, displaying all connectivity providers in your environment.
2. Select the connectivity provider. Click **Remove** to remove the server or cluster as a connectivity provider.

Working with connectivity providers:

You can list, show, and modify a Service Connectivity Management (SCM) connectivity provider using the administrative console.

Procedure

1. Click **Service integration > SCM connectivity providers**. The SCM connectivity providers page opens, displaying all connectivity providers in your environment.
2. Select a connectivity provider to display its details page.
3. Fields can be modified on this page, although you cannot modify the **Name**, **Author**, **Created**, or **Updated** fields.
4. Use the **Apply**, **OK**, **Reset**, and **Cancel** buttons in order to complete any modifications.

Service Connectivity Management usage of Service Component Architecture (SCA) modules:

A Service Component Architecture (SCA) module is installed every time the Service Federation Management console creates a group proxy. You can view SCA modules on the Enterprise Application view and SCA module list of the administrative console.

A versioned SCA module is used for the group proxy. The base module name is `ScmGroupProxy` and the version number is `v1_0_0`. The cell identifier is formed from the connectivity provider name and a unique identifier for the group proxy within the cell.

The name of the service module as it appears in the module list is `ScmGroupProxy (ConnectivityProviderName_UniqueId)`, and the service application name is of the form `ScmGroupProxy_v1_0_0_ConnectivityProviderName_UniqueIDApp`. The same unique identifier also forms part of the URL and Atom identifier used to access the group proxy via the SCM protocol.

A group proxy created on the connectivity provider *ExampleConnectivityProvider* with the generated unique identifier *xot5*, would result in a module with the name `ScmGroupProxy (ExampleConnectivityProvider_xot5)` being deployed as the application `ScmGroupProxy_v1_0_0_ExampleConnectivityProvider_xot5App` to the server or cluster associated with the connectivity provider.

The URL to access the Atom document representing the group proxy resource would be of the form:

```
/rest/scmp/connectivity-provider/ExampleConnectivityProvider-g0jk9fzm/mediation/group-proxy-type/group-proxy/xot5-g0jkja19
```

The Atom identifier for that document would be of the form:

```
urn:wesb-scmp:cell/localhostNode01Cell/connectivity-provider/  
ExampleConnectivityProvider-g0jk9fzm/mediation/group-proxy-type/group-  
proxy/xot5-g0jkja19
```

Note: Attributes of the SCM group proxy appear as promoted properties of the module. These can be viewed via the administration console but must not be modified.

Service Connectivity Management mapping to proxy gateways:

A Service Connectivity Management (SCM) group proxy module is implemented as a proxy gateway within WebSphere ESB

The SCM proxy targets for the group proxy appear as virtual services of the proxy gateway and can be viewed in Business Space powered by WebSphere via the Proxy gateway widget. Properties of the proxy target appear as properties of the virtual service.

Note: Virtual services associated with SCM group proxy modules must not be added, removed, or modified, via the Proxy gateway widget.

Starting the First steps console

After you install WebSphere Enterprise Service Bus, you can use the First steps console to verify the installation, start the Profile Management Tool, access product documentation, or direct elements such as servers and administrative consoles related to individual profiles.

A generic version of the console and a version for each profile in your installation are available. Options on each console are displayed dynamically, depending on features you install and the availability of elements on each operating system. Options might include verifying your installation, starting or stopping the server or deployment manager, accessing the administrative console, starting the Profile Management Tool, and accessing the product documentation.

You will usually want to start the version for the profile. The following sections provide detailed information on starting a First steps console based on its version and the platform used on the system:

- “Starting the First steps console for a profile on Linux, UNIX, and Windows platforms” on page 280
- “Starting the generic version of the First steps console” on page 280

Restrictions: The First steps console might not start if you use Mozilla as your default browser and it is installed in a location containing a space in the path name. To rectify this problem, perform one of these actions:

- Install Mozilla into a location without a space in the path name.
- Alter the registry key to remove the space.
- Temporarily set Internet Explorer as the default browser and then set Mozilla as the default browser. This approach automatically removes the space from the registry key.

Starting the First steps console for a profile on Linux, UNIX, and Windows platforms

Linux **UNIX** **Windows** Start a First steps console for a profile by performing the following steps:

1. Open a command window.
2. Change to the following directory (where *install_root* represents the installation location of the WebSphere ESB profile (*install_root/profiles/profile_name/*):
 - For WebSphere ESB profiles:
 - **Linux** **UNIX** *profile_root/firststeps/esb*
 - **Windows** *profile_root\firststeps\esb*
3. Issue the **firststeps** command to start the console:
 - **Linux** **UNIX** `./firststeps.sh`
 - **Windows** `firststeps.bat`

Fast path:

You can also start a version of the First steps console associated with a profile by performing one of the following tasks:

- Check the First steps console check box on the Profile creation complete or Profile augmentation complete panel at the end of the profile creation or augmentation process.
- **Windows** Click **Start > Programs > IBM > Enterprise Service Bus 7.5 > Profiles > profile_name > First steps**.

Starting the generic version of the First steps console

Start the generic version of the First steps console by performing the following steps.

1. Open a command window.
2. Change to the following directory:
 - **Linux** **UNIX** *install_root/firststeps/esb*
 - **Windows** *install_root\firststeps\esb*The variable *install_root* represents the location of the WebSphere ESB installation on Linux, UNIX, and Windows systems.
3. Issue one of the following commands to start the console:
 - **Linux** **UNIX** `./firststeps.sh`
 - **Windows** `firststeps.bat`

Fast path:

Windows You can also start the generic version of the console on Windows platforms by clicking **Start > Programs > IBM WebSphere > Enterprise Service Bus 7.5 > First steps**.

Updating WebSphere ESB

You can install updates to WebSphere ESB when they are available.

Updating the software interactively

Install updates to software packages you installed using IBM Installation Manager.

Before you begin

By default, Internet access is required unless your repository preferences points to your local update site.

Each installed package has the location embedded for its default IBM update repository. For Installation Manager to search the IBM update repository locations for the installed packages, the preference **Search service repositories during installation and updates** on the Repositories preference page must be selected. This preference is selected by default.

During the update process, Installation Manager might prompt you for the location of the repository for the base version of the package. If you installed the product from DVDs or other media, they must be available when you use the update feature.

See the Installation Manager information center for more information.

Important: If you are updating to V7.5.1 and you had created profiles in an earlier version, those profiles are preserved and you do not need to recreate them.

About this task

You cannot use this procedure to install updates on the underlying IBM DB2 Express. You must update IBM DB2 Express following its normal update process.

Procedure

To find and install product package updates:

1. Close all programs that were installed using Installation Manager before updating.
2. Start Installation Manager. From the Start page of the Installation Manager, click **Update**.

 You can also click **Start > Programs > IBM > *package group name* > Update**. For example, click **Start > Programs > IBM > WebSphere ESB > Update**.

3. If IBM Installation Manager is not detected on your system or if an older version is already installed, then you must continue with the installation of the latest release. Follow the on-screen instructions in the wizard to complete the installation of IBM Installation Manager.
4. In the Update Packages wizard, select the package group containing the product package you want to update or select the **Update all** check box, and then click **Next**. Installation Manager searches for updates in its repositories and the predefined update sites for the software you are updating. A progress indicator shows the search is taking place.
5. If updates for a package are found, then they are displayed in the **Updates** list on the Update Packages page below their corresponding package. Only the latest recommended updates are displayed by default. Click **Show all** to display all updates found for the available packages.

- a. To learn more about an update, click the update and review its description under **Details**.
 - b. If additional information about the update is available, a **More info** link is included at the end of the description text. Click the link to display the information in a browser. Review this information before installing the update.
6. Select the updates that you want to install or click **Select Recommended** to restore the default selections, and click **Next**. Updates that have a dependency relationship are automatically selected and cleared together.
7. On the Licenses page, read the license agreements for the selected updates. On the left side of the Licenses page, the list of licenses for the updates you selected is displayed; click each item to display the license agreement text. If you agree to the terms of all the license agreements, click **I accept the terms of the license agreements**. Then click **Next**.
8. On the Summary page, review your choices before installing the updates.
 - a. If you want to change the choices you made on previous pages, click **Back**, and make your changes.
 - b. When you are satisfied, click **Update** to download and install the updates. A progress indicator shows the percentage of the installation completed.
9. Optional: When the update process completes, a message that confirms the success of the process is displayed near the top of the page. Click **View log file** to open the log file for the current session in a new window. You must close the Installation Log window to continue.
10. Click **Finish** to close the wizard.
11. Close Installation Manager.

Related information:

 IBM Installation Manager information center

Rolling back updates

Using the Roll back packages wizard, you can remove updates to the WebSphere ESB installation and revert to a previous version.

Before you begin

During the rollback process, Installation Manager must access files from the earlier version of the package. By default, these files are stored on your system when you install a package. If the files are not available on your workstation, you must include the location of the repository from which you installed the previous version of the product in your Installation Manager preferences (**File > Preferences > Repository**). If you installed the product from DVDs or other media, they must be available when you use the rollback function.

About this task

Use the rollback function if you have applied an update to a product package, and decide later that you want to remove the update and revert to the earlier version of the product. When you use the rollback function, the Installation Manager uninstalls the updated resources, and reinstalls the resources from the previous version.

When you roll back to an earlier version of a package, it is restored with same features that were associated with that version. Use the Modify Packages wizard to add and remove features.

For more information about Installation Manager, see the Installation Manager information center.

Procedure

1. Close all programs that were installed using Installation Manager before rolling back.
2. Start the Installation Manager.
3. From the Start page of the Installation Manager, click **Roll back** to start the Roll back packages wizard.
4. On the Roll Back Packages page, from the Package Group Name list, select the package group that contains the packages that you want to roll back and click **Next**.
5. Select the version of the package that you want to roll back to and click **Next**.
6. Read the summary information and click **Roll Back** to roll back the package.
7. Optional: When the rollback process completes, a message that confirms the success of the process is displayed near the top of the page. Click **View log file** to open the log file for the current session in a new window.
8. Click **Finish** to close the wizard.
9. Close Installation Manager.

Results

The package you selected to roll back is removed.

Related information:

 IBM Installation Manager information center

Manually installing an interim fix

Manually install an interim fix for WebSphere ESB from a local repository using IBM Installation Manager.

Before you begin

You must log into the system using the same user account that you used to install the product packages.

About this task

Installation Manager works with repositories to install updates. A repository can be an online location that hosts the interim fix files and other configuration information, or a local file system that contains the files. For more information about Installation Manager, see the Installation Manager information center.

To install an interim fix silently (for example, if you do not have access to the Installation Manager launchpad), refer to “Silently installing an interim fix” on page 284.

Procedure

To install an interim fix using Installation Manager, complete the following steps:

1. Download the interim fix to the local system.
2. Create a new directory, for example JR39658.
3. Extract the interim fix in the new directory.
4. Add the new directory to Installation Manager, as follows:
 - a. Start Installation Manager.
 - b. From the Start page, click **File > Preferences > Repositories**.
 - c. From the Repositories page, click **Add Repository**.
 - d. In the Add Repository window, browse to the new directory you created for the interim fix files.
 - e. Select the repository.config file and click **Open**.
 - f. From the Repositories page, click **OK**.
5. From the Installation Manager home page, click **Update**, and then select the WebSphere ESB installation to update. Installation Manager goes through the repository list (including the repository you just added) and provides a list of packages.
6. Select the fix (or fixes) you want to install and then click **Next**.

Results

Installation Manager installs the selected fix or fixes.

Related tasks:

“Silently installing an interim fix”

You can install an interim fix for WebSphere ESB using the command-line mode of the Installation Manager.

Related information:

 IBM Installation Manager information center

Silently installing an interim fix

You can install an interim fix for WebSphere ESB using the command-line mode of the Installation Manager.

Before you begin

You must log into the system using the same user account that you used to install the product packages.

About this task

A repository can be an online location that hosts the interim fix files and other configuration information, or a local file system that contains the files. This procedure uses a command to specify the local directory of the interim fix. To install an interim fix using the Installation Manager launchpad, refer to “Manually installing an interim fix” on page 283.

Procedure

To silently install an interim fix, complete the following steps:

1. Download the interim fix to the local system.

2. Create a new directory and extract the interim fix in the new directory.
3. Open a command prompt, and change directories to the `/eclipse/tools` directory under Installation Manager.

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

4. Make the appropriate replacements and run the following command:

```
imcl install fixID -repositories repositoryLocation -installationDirectory installationDirectory -log logLocation
```

- a. Replace *fixID* with the ID of the interim fix. The ID can be found in the `repository.xml` file in the directory where you extracted the interim fix, in the `fix id` element. For example:

```
<fix id="7.5.1.0-WS-BPMADVWESB-IFJR39658" version="0.0.0.20111115_1047" offeringId="EnhancedFix" offeringVersion="0.0.0.EnhancedFix">
```

- b. Replace *repositoryLocation* with the directory where you extracted the interim fix.
- c. Replace *installationDirectory* with the location where you installed WebSphere ESB.
- d. Replace *logLocation* with the location and file name to log the installation information.

For example:

```
C:\Program Files\IBM\Installation Manager\eclipse\tools>imcl install 7.5.1.0-WS-BPMADVWESB-IFJR39658 -repositories
C:\interimFix\7.5.1.0-WS-BPMADVWESB-IFJR39658/ -installationDirectory C:\IBM\BPM75 -log logfix.txt
```

Results

The installation log (specified by the **-log** parameter) contains no error messages if the interim fix installation is successful. The command line shows a message that the fix was installed. For example:

Installed 7.5.1.0-WS-BPMADVWESB-IFJR39658_0.0.0.20111115_1047 to the C:\IBM\BPM75 directory.

Related tasks:

“Manually installing an interim fix” on page 283

Manually install an interim fix for WebSphere ESB from a local repository using IBM Installation Manager.

Related information:

 IBM Installation Manager information center

Silently uninstalling an interim fix

You can uninstall an interim fix for WebSphere ESB using the command-line mode of the Installation Manager.

Before you begin

You must log into the system using the same user account that you used to install the product packages.

Procedure

To silently uninstall an interim fix, complete the following steps:

1. Open a command prompt, and change directories to the `/eclipse/tools` directory under Installation Manager.

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

2. Make the appropriate replacements and run the following command:

```
imcl uninstall fixID -installationDirectory installationDirectory -log logLocation
```

- a. Replace *fixID* with the ID of the interim fix. The ID can be found in the repository.xml file in the directory where you extracted the interim fix, in the fix id element. For example:

```
<fix id="7.5.1.0-WS-BPMADVWESB-IFJR39658" version="0.0.0.20111115_1047" offeringId="EnhancedFix" offeringVersion="0.0.0.EnhancedFix">
```

- b. Replace *installationDirectory* with the location where you installed WebSphere ESB.
- c. Replace *logLocation* with the location and file name to log the information.

For example:

```
C:\Program Files\IBM\Installation Manager\ eclipse\tools>imcl uninstall 7.5.1.0-WS-BPMADVWESB-IFJR39658 -installationDirectory C:\IBM\BPM75 -log logfix.txt
```

Results

The log (specified by the **-log** parameter) contains no error messages if uninstalling is successful. The command line shows a message that the fix was uninstalled.

Uninstalling WebSphere ESB

You can remove WebSphere ESB interactively or silently.



Uninstalling WebSphere ESB interactively

The Uninstall option in the Installation Manager enables you to uninstall packages from a single installation location. You can also uninstall all the installed packages from every installation location.

About this task

To uninstall the packages, you must log in to the system using the same user account that you used to install the product packages. A package cannot be uninstalled when another package has a dependency on it, unless the dependent package is also selected to be uninstalled.

Procedure

1. Close the programs that you installed using Installation Manager.
2. Stop all running servers.
3. Start the Installation Manager. On the Start page, click **Uninstall**.  On Windows, you can also click **Start > Programs > WebSphere Enterprise Service Bus > IBM WebSphere > Enterprise Service Bus 7.5 > Uninstall**.
4. On the Uninstall Packages page, select WebSphere Enterprise Service Bus and associated packages and click **Next**.  If you selected **Start > Programs > IBM WebSphere > Enterprise Service Bus 7.5 > Uninstall** in the previous step, WebSphere Enterprise Service Bus is pre-selected for uninstallation on the Uninstall Packages page.
5. On the Summary page, review the list of packages that will be uninstalled and then click **Uninstall**. After the uninstallation finishes, the Complete page opens.
6. Click **Finish** to exit the wizard.

Results

When WebSphere Enterprise Service Bus is uninstalled, all profiles that are augmented to WebSphere Enterprise Service Bus are removed, including any WebSphere Application Server profiles that are augmented to WebSphere Enterprise Service Bus.

What to do next

If you plan to reinstall WebSphere Enterprise Service Bus, and databases were created in the previous install, the databases must be dropped before you can create a new profile. See [Reinstallation cannot create new profile](#).

Linux If you plan to reinstall WebSphere Enterprise Service Bus, you must delete the remaining DB2 Express entries in the `/etc/service` file. This is necessary because the new installation requires that port 50000 be available. Search the `/etc/service` file and remove any references to DB2 Express and port 50000. For example:

```
db2c_bpminst 50000/tcp
```

or

```
db2c_db2inst1 50000/tcp
```

Uninstalling WebSphere ESB silently

You can use the command-line mode of the Installation Manager to uninstall WebSphere ESB.

Before you begin

Close all programs that you installed using the Installation Manager.

About this task

To uninstall, you must log in to the system using the same user account that you used to install.

Procedure

To silently uninstall WebSphere ESB, complete the following steps:

1. Open a command prompt, and change directories to the `/eclipse/tools` directory under Installation Manager.

Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

2. Make the appropriate replacements and run the following command:

```
imcl uninstall list_of_product_IDs -installationDirectory installationDirectory -log logLocation
```

- a. Replace `list_of_product_IDs` with a list of the IDs for the products you want to uninstall, separated by spaces.

Table 85. Product IDs

| Product | Product ID |
|---------------|-------------------|
| WebSphere ESB | com.ibm.ws.WESB75 |

Table 85. Product IDs (continued)

| Product | Product ID |
|---|--|
| WebSphere Application Server Network Deployment | com.ibm.websphere.ND.v70,core.feature,samples,import,productProviders,fe (includes all required features) |
| Feature Pack for Service Component Architecture (SCA) | com.ibm.websphere.SCA.v10 |
| Feature Pack for XML | com.ibm.websphere.XML.v10 |
| Installation Manager | com.ibm.cic.agent,agent_core,agent_jre |
| DB2 for Linux 32-bit | com.ibm.ws.DB2EXP97.linuxia32 |
| DB2 for Linux 64-bit | com.ibm.ws.DB2EXP97.linuxia64 |
| DB2 for Windows 32-bit | com.ibm.ws.DB2EXP97.winia32 |
| DB2 for Windows 64-bit | com.ibm.ws.DB2EXP97.winia64 |

- b. Replace *installationDirectory* with the location where you installed the product.
- c. Replace *logLocation* with the location and file name to log the information.

Results

Installation Manager uninstalls the list of products and writes a log file to the directory that you specified.

Example

The following example uninstalls WebSphere ESB from Windows.

C:\Program Files\IBM\DW\Install\Uninstall>uninstall -com.ibm.ws.WS70 -com.ibm.websphere.ND.v70,core.feature,samples,import,productProviders,fe -com.ibm.ws.WS70 -com.ibm.websphere.SCA.v10 -com.ibm.ws.XML.v10 -com.ibm.cic.agent,agent_core,agent_jre -com.ibm.ws.DB2EXP97.linuxia32 -com.ibm.ws.DB2EXP97.linuxia64 -com.ibm.ws.DB2EXP97.winia32 -com.ibm.ws.DB2EXP97.winia64 -installDir installationDirectory C:\IBM\DW\Uninstall\log.txt

Chapter 2. Migrating from earlier products and versions

Migrating refers to the process of moving applications and configuration information from an older version of a product to a later version of the product or from one product to a different product.

Note: The Process Designer, Process Center, and each runtime environment must be running the same version of WebSphere ESB.

Before you begin migrating, check the support site and make sure that you have the latest available fixes. See the link in Related reference.

Related information:

 PDF documentation

WebSphere Enterprise Service Bus documentation (in PDF format)

 Information roadmaps

Information roadmaps on IBM developerWorks organize information about WebSphere Business Process Manager, WebSphere ESB, and the other products in the portfolio.

 IBM Education Assistant

Multimedia educational modules about WebSphere ESB, provided by IBM Education Assistant.

 Technotes

WebSphere ESB Support > technote search for 7.5 migration category documents. Use the document type, product category, and search terms fields to find the information you need.

 Overview

Overview tab, on product library Web page. Use this page to access announcements, data sheets, and other general library documents related to WebSphere ESB.

Migration overview

The process of moving applications, configuration, and databases from an earlier version of WebSphere ESB to this version of WebSphere ESB is referred to as version-to-version migration, or simply migration.

Related tasks:

“Migrating a network deployment environment with full downtime” on page 322
Use this procedure to migrate a network deployment environment while incurring full downtime.

“Migrating a network deployment environment with minimal downtime” on page 330

Use this procedure to migrate a network deployment environment while incurring minimal downtime.

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

What is version-to-version migration?

Version-to-version migration refers to the movement of profiles, applications, and data associated with an earlier version of WebSphere ESB to a newly installed version of WebSphere ESB.

Version-to-version migration overview

Version-to-version migration, or simply migration, refers to the process of moving applications that have been developed on prior releases to a newer version of WebSphere ESB. Migration can be accomplished using a set of migration facilities provided by IBM Integration Designer for migrating applications, or migration can be accomplished in a production environment using a set of runtime migration procedures and tools for migrating the entire production configuration, applications, and databases.

In IBM Integration Designer, applications and workspaces developed using earlier versions can be imported and migrated to V7.5.1. Once the applications have been migrated to V7.5.1, they can either be directly deployed to a V7.5.1 runtime, or they can be enhanced to exploit new capability delivered in V7.5.1 and then be deployed. This style of migration is referred to as artifact migration.

The migration of applications deployed to production environments goes well beyond the scope of redeployment of the applications to the new version. The configuration of the production topology, the product databases, and the product data in the databases are all migrated to V7.5.1 by a consistent set of procedures and tools. The process associated with the set of procedures and tools for migrating production configuration, applications, and databases is referred to as *runtime migration*.

The products leveraging the common runtime migration procedures and tools include:

- WebSphere Process Server
- WebSphere Enterprise Service Bus
- IBM Business Monitor

Applications can also be manually redeployed from a production environment that is the source of the migration to a parallel target production environment. This style of migration is referred to as manual migration.

Product updates

The version-to-version migration process differs from the process of applying interim fixes and updates to the development and production environments. For

information on updates in the forms of interim fixes, fix packs, and refresh packs, see the "updating" topic for your WebSphere ESB product.

WebSphere ESB Migration roadmap

The WebSphere ESB migration roadmap shows the high-level tasks involved in a version-to-version migration.

Use the following flow diagram and high-level migration task descriptions to learn about the tasks involved in a version-to-version migration.

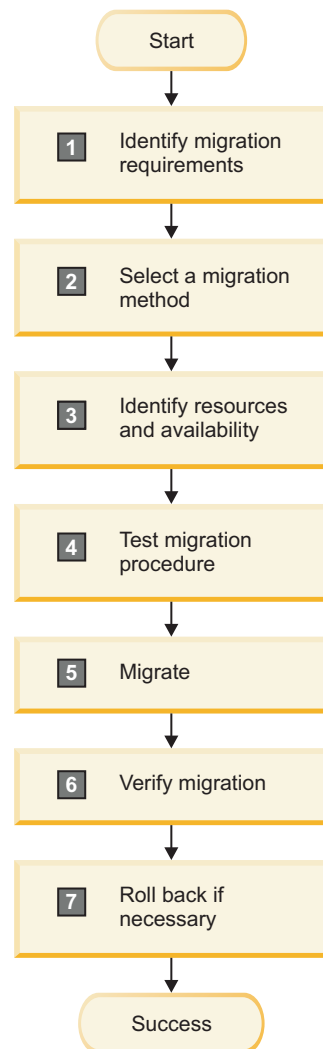


Figure 9. WebSphere ESB migration roadmap for version-to-version migration

1 Identify the migration requirements

Identifying the migration requirements is the first step in planning your migration.

See the Migration method comparison topic for a list of the set of considerations involved in the migration process.

2 Select a migration method

There are three migration methods to choose from when migrating:

- Runtime migration

- Manual migration
- Artifact migration

To review the migration methods and determine which migration method fits your requirements see the Migration methods topic.

3 Identify resources and availability

When planning your migration, it is critical to identify the availability of all of the resources you need for the migration, including:

- Human resources: How many people and what skill level is needed? What is the timeframe for the human resource need?
- Hardware and software resources: What hardware or software do you need to acquire to ensure a successful migration?

Depending on your WebSphere ESB configuration, visit:

- IBM WebSphere Enterprise Service Bus system requirements

4 Test the migration procedure

Before performing the migration, thoroughly test the migration procedure:

- Test your applications in a new environment.
- Test your migration procedure in a staging environment.
- Practice your rollback plan on a test system.

When planning which test or tests will best suit your migration, keep in mind the necessary resources to make the test successful.

5 Migrate

Use the migration procedures associated with the migration method you choose to migrate your environment.

6 Verify migration

After performing the migration, use one of the following methods to verify success, depending on which migration method you used:

- If you used the runtime migration method, see Verifying migration.
- If you used the manual migration method, manually redeploying your applications from a production environment that is the source of the migration to a parallel target production environment, verify that your applications work as expected.
- If you used the artifact migration method, using IBM Integration Designer and IBM Business Modeler to import and update applications and workspaces developed using earlier versions, verify that your applications work as expected.

7 Roll back if necessary

If the migration was not successful, you might need to roll back your environment and perform the migration again. Use one of the following rollback methods, depending on which migration method you used:

- If you used the runtime migration method, see Rolling back your environment.
- If you used the manual migration method, you might need to uninstall and then reinstall the applications.
- If you used the artifact migration method, you might need to uninstall and then re-import and re-migrate the applications and source artifacts using Integration Designer or IBM Business Modeler.

Migration methods

There are three types of version-to-version migration methods to choose from when considering moving to a new version of WebSphere ESB: runtime migration, manual migration, and artifact migration.

- “Runtime migration (production environment)”
- “Manual migration (parallel production environment)”
- “Artifact migration (parallel production environment with development tool migration)” on page 294

Runtime migration (production environment)

In production environments, you can use runtime migration procedures and tools to migrate topology configuration, applications, and databases to the new version of WebSphere ESB. The runtime migration procedures and tools support both stand-alone and network deployment environment migrations, as well as variants that include migration to a remote system (stand-alone environments only), migration while an operating system is being upgraded to a supported version (stand-alone environments only), and network deployment variants to support full downtime migration windows and minimal downtime migration windows. The runtime migration process replicates the source production configuration into the target environment. During the migration process, the target production environment replaces the source production environment, so the two environments are never operated in parallel.

The runtime migration procedures and tools should be used in the following scenarios:

- You want to move your applications to the new version without a dependency on the development tools and the development environment.
- You want to have your source production environment configuration and applications automatically replicated in the target production environment.
- You have product data in queues or failed events in product databases that were created in the source environment and need to survive the migration and be managed in the target production environment.
- You can tolerate a production environment downtime window to perform the migration.

The high-level tasks involved in runtime migration are:

1. Install the new product version.
2. Back up all production profiles and databases.
3. Migrate each source environment profile to the target environment.
4. Migrate or upgrade the product databases.
5. Migrate the product database data.

For more information on the runtime migration procedures and tools, see the “Migration overview” on page 289 topic.

Manual migration (parallel production environment)

An alternative to using the migration procedures and tools is to use the manual version-to-version migration process. With the manual migration process, you are free to create a parallel target production environment that is configured from scratch differently from the source production environment. Applications can then

be selectively redeployed from the source production environment to the target production environment. The redeployed applications create their own database tables and application data in the parallel production environment so they do not have access to the application data stored in the databases configured for the source production environment.

The manual runtime migration process should be used in following scenarios:

- You want to move your applications to the new version without depending on the development tools and the development environment.
- You want to reconfigure your topology as part of the process of migrating to the new version of WebSphere ESB.
- You have application data in queues or failed events in product databases that were created in the source environment that can be managed to completion in the source production environment while new messages and events are routed in parallel to the target production environment.
- You cannot incur any downtime in your production environment and can concurrently manage parallel source and target production environments.
- You want to selectively redeploy applications from your source production environment to your target production environment.

The high-level tasks involved in manual migration are:

1. Install the new product version.
2. Configure your desired parallel production environment.
3. Manually deploy applications using the previous version EAR files from the source environment. You can perform the application deployment using the administrative console on the target environment.
4. Optional: Run both environments in parallel.

Artifact migration (parallel production environment with development tool migration)

The artifact migration process is similar to the manual migration process in terms of the configuration of the parallel target production environment, but instead of the applications being manually redeployed from the source environment directly into the target production environment they are imported into the development environment and migrated by the development tools. This results in applications whose artifacts are migrated to the new version, enabling the applications to then be modified to exploit the new capabilities delivered by the new version of WebSphere ESB. The application can then be tested and deployed to the parallel target production environment. Consistent with the manual migration process, when the applications are deployed to the target production environment, they create a new set of database tables, so they do not have access to the application data stored in the databases configured for the source production environment.

The artifact migration should be used in the following scenarios:

- You want to leverage the development tools and development environment to migrate the application artifacts to the new version and validate the compatibility of your applications.
- You want to leverage the development tools to update your applications to exploit new capability delivered by WebSphere ESB.
- You want to reconfigure your topology as part of the process of migrating to the new version of WebSphere ESB, or you can manually duplicate your source production environment configuration in your parallel production environment.

- You have application data in queues or failed events in product databases that were created in the source environment that can be managed to completion in the source production environment while new messages and events are routed in parallel to the target production environment.
- You cannot incur any downtime in your production environment and can concurrently manage parallel source and target production environments.
- You want to selectively migrate applications from your source production environment to the new version of WebSphere ESB with the development tools and selectively deploy those applications to your target production environment.

The high-level tasks involved in artifact migration are:

1. Install the new product version.
2. Configure your desired parallel production environment.
3. Import the applications from the source production environment into development tools and migrate the applications according to the development tool's migration procedures.
4. Optional: Update the migrated applications to exploit new capability delivered in WebSphere ESB.
5. Manually deploy the migrated applications from the development tools to the target production environment.
6. Optional: Run both environments in parallel.

Migration method comparison

To determine the most appropriate migration method for migrating to a newer version of WebSphere ESB, analyze the amount of stateful data in the environment, the amount of downtime your system can support, and whether you want to preserve your previous configuration.

Migration method considerations

There are several different issues to consider when determining the right migration method for migrating to a newer version of WebSphere ESB. The following section enumerates a set of items to consider when deciding which method best fits your migration requirements.

- Production data
- Downtime
- Application enhancements
- Target environment configuration
- Risk mitigation
- Selective or phased application migration

Production data

The runtime migration method results in the source production environment being replaced by the target production environment. The implication on application data is that data that was created in the database by the source environment is available to the target environment post migration. Messages in queues and failed events that existed in the source environment can be managed by the target environment post migration. The runtime migration method is the only method that provides this capability. The manual and artifact migration methods both result in a parallel production environment that has its own separate databases

configured, completely distinct and independent of the source environment, even when the applications from the source environment are deployed to the target environment.

Downtime

The runtime migration method results in the source environment being replaced by the target environment while the manual and artifact migration processes depend on the creation of a parallel target environment. The implication is that the runtime migration method requires a downtime period when the databases are being upgraded and migrated from the source version to the target version prior to starting the migrated servers. The runtime migration procedures provide a minimal downtime procedure that can be used in some cases, but still does not eliminate the need for downtime.

The manual and artifact migration methods both require a parallel environment to be created that can be used in production concurrently with the source environment. The source and target environments can execute side-by-side until it is appropriate for the source environment to be discontinued. The ability to have two environments running concurrently on different versions also implies a level of operational complexity and likely requires additional capacity.

Long-running processes and human tasks

Application enhancements

The advantage of using artifact migration and the development tools is that the applications can be updated to the newer version artifact level and then be enhanced with features provided in the newer version.

Target environment configuration

If you require the same configuration in your target environment as your source environment, the runtime migration method is typically more appropriate because it will automatically replicate the source environment's topological configuration to the target environment. However, if you need to reconfigure the target environment configuration completely differently than your source environment for one of several good reasons, you must either do that before or after version-to-version migration as an independent exercise, or use either the manual or artifact migration methods if you plan to do it concurrent with the version-to-version migration.

Risk mitigation

The parallel environments provided by the manual and artifact migration methods enable a target production environment that is completely independent of the source environment that is serving the existing consumers enabling the target environment to be rigorously tested before going live in a production setting. In addition, artifact migration can reduce risk by leveraging the development tools to aid in verification that the application being migrated does not contain any issues that would present backwards compatibility challenges. Even in scenarios where migrations are leveraging the runtime or manual migration methods, artifact migration validation using the development tools is often done as an initial stage of the migration effort to validate application compatibility.

Selective or phased application migration

If you have a situation where you do not want to migrate all your applications in a single downtime window to the target version, you should use either the manual or artifact migration approaches. These

approaches provide support for two parallel environments, the source and the target, and support selective or phased deployment of the migrated applications to the target environment. In contrast, the runtime migration method migrates all applications from the source environment to the target environment.

Migration method comparison

Use the following table to compare the benefits, costs, and risks of the three migration methods:

Table 86. Version-to-version migration methods: a comparison

| Migration method | Benefits | Costs | Risks |
|-------------------|--|--|---|
| Runtime migration | <ul style="list-style-type: none"> • No dependency on the development tools • Source environment configuration is replicated in the target environment • Source environment applications are migrated to the target environment • Source environment application data is moved, using existing database tables • Application instance data on queues and failed events in the source environment can be handled post migration by the target environment • Additional hardware and/or software resources not required to manage another production environment | <ul style="list-style-type: none"> • Downtime is required when the target product environment assumes the role of the source production environment • Requires all applications on a node to be ready to migrate at the same time • New features are not enabled automatically and sometimes unavailable without migrating the application artifacts using artifact migration • Parallel production environment cannot be set up • Test focus: <ul style="list-style-type: none"> – End-to-end testing to validate migration process – Regression testing and performance tuning | <ul style="list-style-type: none"> • A rollback plan must be in place to handle a possible migration failure. For more information, see Rolling back your environment. • Existing user applications should continue to execute in the new runtime at the same level of function they had in the old runtime. In some cases, however, there may be a change in code on which the application depends, such as a JDK change, which may have negative impact on the unchanged application. |

Table 86. Version-to-version migration methods: a comparison (continued)

| Migration method | Benefits | Costs | Risks |
|--------------------|---|--|--|
| Manual migration | <ul style="list-style-type: none"> • No dependency on the development tools • Target production environment can be configured differently than the source production environment since configuration is not automatically migrated from the source to the target • Parallel production environment supported: <ul style="list-style-type: none"> – Selective application migration – No downtime • Ability to perform extensive testing before migrating to production environment, but usually regression testing is enough • No dependency on migration tools | <ul style="list-style-type: none"> • Existing data is not moved; new database tables are created • New features are not enabled automatically and sometimes unavailable without migrating the application artifacts using artifact migration • Manual (scripted) deployment of applications is required • Requires updates to client applications • Hardware and software licenses may need to be evaluated for any additional licenses required when running in parallel | <ul style="list-style-type: none"> • Existing user applications should continue to execute in the new runtime at the same level of function they had in the old runtime. In some cases, however, there may be a change in code on which the application depends, such as a JDK change, which may have negative impact on the unchanged application. |
| Artifact migration | <ul style="list-style-type: none"> • Ability to exploit new features • Parallel production environment supported: <ul style="list-style-type: none"> – Selective application migration – No downtime • Ability to perform extensive testing before migrating to production environment • No dependency on migration tools | <ul style="list-style-type: none"> • New development environment is required • Existing data is not moved; new database tables are used • Manual (scripted) deployment of applications is required • Requires updates to client applications • Hardware and software licenses may need to be evaluated for any additional licenses required when running in parallel • Additional test coverage for application updates is required | <ul style="list-style-type: none"> • Application updates might require some level of testing. |

Supported source migration paths

The following product and version combinations are supported as sources for version-to-version migrations to WebSphere ESB V7.5.1.

- WebSphere ESB version 7.5
- WebSphere ESB version 7.0.0.x
- WebSphere ESB version 6.2.0.x
- WebSphere ESB version 6.1.0.x

Note: If you are migrating from a version of the product that is earlier than V6.1.0.x, you must first migrate to one of the versions that are supported migration sources using the manual migration method, and then you can use the runtime migration method to migrate from that version to V7.5.1.

Migration types

Runtime migration supports the migration of stand-alone environments and network deployment environments.

Stand-alone migration

The following types of stand-alone migration variants are supported by the runtime migration procedures and tools:

- **Side-by-side migration:** where the source and target of the migration are on the same system
- **Remote migration:** where the source and target of the migration are on different systems
- **Operating system upgrade migration:** where the operating system on the source system is being upgraded during the migration procedure to a new version that is supported by WebSphere ESB V7.5.1.

The following sections describe each of these types of stand-alone environment migration variants in more detail.

Stand-alone side-by-side migration

The stand-alone side-by-side migration process is the simplest runtime migration scenario where the target product is installed on the same system as the source product, and the runtime migration procedures and tools are used to migrate the stand-alone profile containing the configuration, applications and the product databases to the target environment.

Stand-alone remote migration

The stand-alone remote migration process enables WebSphere ESB V7.5.1 to be installed on a different system from the source of the migration in order to support migration of the configuration and applications from one system to another. The stand-alone remote migration process can be used to support a variety of scenarios including:

- Migrating to a remote system that has the same type of hardware, operating system, and operating system version as the source of the migration
- Migrating to a remote system that has a different type of hardware (64-bit for example), a different operating system, or a different operating system version

The process requires the migration commands on the target system to be copied to the source system where they are used to copy the source profiles. The snapshot directory is then copied to the target system and used as the source for the profile migration.

Stand-alone operating system upgrade migration

The stand-alone operating system upgrade migration process enables the operating system on the system containing the source of the migration to be upgraded during the migration process. This is typically necessary if the operating system version containing the source product version is no longer supported by WebSphere ESB V7.5.1.

The process requires that you copy each of the source profiles on the prior version of the operating system, back up the copied source profiles to a remote location, reinstall the operating system to the new version, install the target product, restore the copied source profiles back to the migration system with the updated operating system, and then use the snapshot directory as the source for the profile migration.

Network deployment migration

Network deployment environment migrations are more involved than stand-alone environment migrations due to the need to migrate the deployment manager, clusters, nodes, and differently scoped product databases, in the appropriate order. The side-by-side migration, remote migration, and operating system upgrade migration are also supported for network deployment environments. All network deployment migrations require WebSphere ESB V7.5.1 to be installed side-by-side with the source product of the migration or a different system. If the source of the migration is augmented by additional BPM products, they should be installed in the same installation directory as WebSphere ESB V7.5.1.

Two different types of nodes are referred to in the network deployment migration procedures, **clustered nodes** and **non-clustered managed nodes**. Clustered nodes contain at least one server that is a member of a cluster. Non-clustered managed nodes do not contain any servers that are cluster members.

Runtime migration tools

Migrating stand-alone and network deployment environments requires that you manage the production environment (start and stop the deployment manager, servers, and nodes), migrate configuration profiles, upgrade product databases, and migrate application data. The runtime migration procedures guide you through the process and the runtime migration tools are used to perform the required steps.

The following three sets of tools support the runtime migration procedures:

- “Profile migration tools”
- “Database upgrade and migration tools” on page 301
- “WebSphere Application Server management tools” on page 301

The following sections provide a summary of each of these groups of tools.

Profile migration tools

The profile migration tools are used to migrate the profiles that contribute to the cell, clusters, non-clustered managed nodes, or stand-alone servers being migrated.

The profile migration tools support a three-step process for each profile:

1. Snapshot the configuration files from the source profile to be migrated.
2. Create the target profile in the target installation using the snapshot configuration from the source profile.
3. Migrate the configuration snapshot to the target profile.

The three-step process required to migrate each profile is supported by the WebSphere ESB profile migration wizard that can be invoked via the BPM Migrate command-line utility or the following set of profile migration command-line tools:

- BPMSnapshotSourceProfile command-line utility

- BPMCreateTargetProfile command-line utility
- BPMMigrateProfile command-line utility

Note: The WebSphere ESB profile migration wizard is supported on the following platforms:

- Windows x86 (32-bit)
- Windows x64 (64-bit)
- Linux x86 (32-bit)
- Linux x86-64 (64-bit)
- Linux PPC (32-bit only)
- AIX PPC (32-bit only)
- Solaris SPARC (32-bit only)
- HP-Unix IA64 (64-bit)

Other platforms must use the command-line tools instead of the WebSphere ESB profile migration wizard to perform the three-step profile migration process.

In addition to the three-step process for profile migration, the following command-line utilities play key roles in profile migration:

- The BPMCreateRemoteMigrationUtilities command-line utility creates an archive that can be copied to source migration systems to support remote profile migration.
- The BPMMigrateCluster command-line utility is required in addition to the profile migration tools to migrate cluster profile configuration information in a network deployment environment.

Database upgrade and migration tools

WebSphere ESB V7.5.1 uses the following product databases that are either automatically or manually upgraded or migrated during the migration of the environment:

- Business Space database
- Common database
- Common Event Infrastructure database
- Messaging Engine database

The Common Event Infrastructure database and the Messaging Engine databases are both automatically migrated as needed by the profile migration process. For manually updating the product databases, the commands and scripts for each of the supported database types must be invoked on the database system by a user with sufficient privileges or a system that has the database client utilities installed with a network connection to the database system. The runtime migration procedures describe how to copy the commands and scripts for your database type and the source release of the migration to the database system.

WebSphere Application Server management tools

During the migration procedures, the deployment manager, nodes, and servers must be stopped and started at various steps. In addition, there are several other WebSphere Application Server commands that are used throughout the migration procedures.

Related reference:

 BPMSnapshotSourceProfile command-line utility

 BPMMigrateProfile command-line utility

Related information:

 BPMCreateTargetProfile command-line utility

Profiles

The runtime migration tools support the migration of WebSphere Enterprise Service Bus and WebSphere Application Server source profiles to the same profile type on the migration target.

WebSphere Enterprise Service Bus profile

A WebSphere Enterprise Service Bus profile is one that WebSphere Application Server created using one of the following profile templates: “default.esbserver”, “dmgr.esbserver”, or “managed.esbserver.” When using the Profile Management Tool (PMT), this means that you select **WebSphere ESB** on the Augment Selection page.

WebSphere Application Server profile

A WebSphere Application Server profile is one that WebSphere Application Server created using one of the following profile templates: “default”, “dmgr”, or “managed.” When using the Profile Management Tool (PMT), this means that you select **WebSphere Application Server** on the Environment selection page.

Important: Even though the above definitions refer to the Profile Management Tool as a tool that may have been used to create the source profiles being migrated, you cannot use the Profile Management Tool or the **manageprofiles** command-line utility to create profiles that are the target of a migration.

The runtime migration procedures require the use of the Profile Migration Wizard or the BPMCreateTargetProfile command-line utility to create the migration target profiles.

Deployment manager profile

In a network deployment environment, the deployment manager must be created using the WebSphere ESB deployment manager profile.

Product profile augmentation

The runtime migration tools support the migration of source profiles that have been augmented by one or more of the following products:

- WebSphere Process Server
- WebSphere Enterprise Service Bus
- WebSphere Business Monitor

Augmented source profiles are migrated to a target profile that is augmented with the same product profiles, so the target installation must have at least the same profile capabilities as the source.

For example, if a source installation contains a managed profile that has been augmented by WebSphere ESB and WebSphere Business Monitor, the target installation directory must contain both WebSphere ESB and IBM Business Monitor. In this scenario, the Profile Migration Wizard or the BPMCreateTargetProfile command-line utility will create a target profile that is augmented by WebSphere ESB and IBM Business Monitor.

In a multi-product augmentation environment, where a cell may have clusters and nodes within profiles at various augmentation levels, the deployment manager profile must be augmented at the same augmentation level as the highest augmentation level of any of the profiles in the clusters or nodes.

Mixed-version environments

Version-to-version migration of network deployment-based production environments frequently results in a period of time when the network deployment environment is running applications on different versions of WebSphere ESB. This concept is referred to as **mixed versions**.

Mixed versions of a product can theoretically be applied to multiple cells, mixed-version cells (multiple clusters or managed non-clustered nodes in a single cell), or mixed-version clusters (managed nodes in a single cluster). Only two of these types mixed versions are supported by WebSphere ESB: **multiple cells** and **mixed-version cells**.

Multiple cells

If you have two cells that are initially at version 6.x or 7.0, one can be upgraded to V7.5.1 without having any administrative or database impact on the other cell. This is the simplest way to manage applications that are frequently running on different versions of WebSphere ESB.

Mixed-version cells

In addition to having cells at different versions, clusters and non-clustered managed nodes in a single cell can be at different versions. For example, a cell might have one cluster at version 6.x or 7.0 and another cluster that was at version 6.x or 7.0 that has been migrated to V7.5.1. In a mixed-version cell environment, the cell-scoped Common database is being shared by all the clusters and non-clustered managed nodes that are running different versions of WebSphere ESB.

Mixed-version clusters

WebSphere ESB does not support nodes in a single cluster running on different versions of WebSphere ESB. This concept is referred to as a mixed-version cluster. If you have configured a cluster containing servers running different versions, all the members running earlier versions of the product must be stopped before you start the first V7.5.1 cluster member. Also, once the V7.5.1 cluster member is started, the members of the cluster configured at a pre-V7.5.1 level must not be started.

Databases

WebSphere ESB uses several product databases that are either automatically migrated or must be manually migrated as part of the runtime migration procedure.

Database scopes

Some of the WebSphere ESB product databases are cell-scoped and others are deployment target-scoped.

Common database

This database is cell-scoped, so when a deployment target such as a server, cluster, or non-clustered managed node in the cell is migrated to version 7.5.1, the database must also be migrated. In a mixed-version cell environment, this might result in pre-version 7.5.1 servers, clusters, and non-clustered managed nodes using the same instance of the database as version 7.5.1 servers, clusters, and non-clustered managed nodes.

Note: If you are migrating from version 7.0 to 7.5.1, it is not necessary to perform a schema upgrade for the Common database. Refer to “Migrating databases” on page 356.

Business Space database, Common Event Infrastructure database, and the Messaging Engine database

These databases are deployment target-scoped. They can be configured at a server or a cluster scope. If these databases are configured in a mixed-version cell environment, each server, cluster, or non-clustered managed node will have a unique instance of the database, and each instance will have schema and data that are unique to that version of the product. When each server, cluster, or non-clustered managed node is migrated, its unique instance of the database is migrated also as part of the runtime migration procedures.

Backups

The migration procedures include steps for backing up the product databases to enable them to be restored if schema migration or data migration fails.

Automatic and manual migration

The Common Event Infrastructure database and Messaging Engine database are automatically migrated by the runtime migration procedure when the profiles are migrated.

The Common database is automatically migrated in some situations as part of the runtime migration procedure and in other conditions manual migration is necessary. The Business Space databases require manual migration in all circumstances. In summary, you must update the databases manually using scripts provided with WebSphere ESB in the following circumstances:

- If the server process does not have sufficient permissions (that is, if it has not been configured with a user ID with sufficient permissions for the Common database)
- If you used non-default table spaces
- If your migration source is configured with Business Space

More details on when and under what conditions the product databases should be manually migrated are captured directly in the runtime migration procedures.

Authorization

Because each of the database scripts require different database permissions, check whether you will be able to run all scripts using a single user ID, or whether your database administrator might have to run any of them.

- For the Common database scripts:

The following permissions are required:

- CREATE TABLE
- ALTER TABLE
- DROP INDEX
- CREATE INDEX
- CREATE VIEW
- DROP VIEW
- CREATE SEQUENCE

Time requirements and tuning options

Depending on the quantity of data and the power of your database server, the data migration step (excluding the time required to backup the database and upgrade the database schema) can take several hours.

DB2 for z/OS and OS/390 Version 7

If you use DB2 for z/OS and OS/390 Version 7, and have not yet upgraded the database to DB2 for z/OS version 8 or DB2 9 for z/OS, you will be asked to do that as part of the runtime migration procedure.

Oracle 9i and the Oracle JDBC driver

If you are using Oracle 9i, and have not yet upgraded your database to 10g or 11g, you will be asked to do that as part of the runtime migration procedure.

If you are using the Oracle ojdbc14.jar or the ojdbc5.jar JDBC driver, you will be asked to install and configure the ojdbc6.jar JDBC driver as part of the runtime migration procedure.

Running SQL upgrade scripts

A database administrator (DBA) can run an SQL upgrade script by invoking the upgradeSchema.bat or the upgradeSchema.sh script or by running the SQL script directly.

Important: The scripts must be run in the following sequence:

1. Run all upgradeTablespac* scripts before executing any upgradeSchema* scripts.
2. Run the upgradSchema_SchemaStatus.sql script before running any other "upgradeSchema*" SQL scripts.

For most databases, including Oracle, the options are embedded in the SQL scripts. Additional options are not required. The following options are strongly recommended for specific database types. These options help in problem determination if the scripts do not work successfully, or if the scripts have to be re-run.

To run the SQL scripts directly, use the following parameters and commands.

DB2

```
db2 -s -t -v -z <log name> -f <script_name>
```

Options:

- s Specifies that the script is to exit as soon as the first error occurs
- v Specifies that the statement is to be printed on the command line so that it can be logged in a log file
- z Specifies that the output from the command is to be dumped into a log file that the database administrator (DBA) can use for error checking
- t Specifies that the semicolon (;) is to be treated as a delimiter for the commands in the file
- f Specifies a file name

SQLServer

```
osql -e -b -U <username> -P <password> -i <script_name> -o <log name>
```

Options:

- e Specifies that the command is to be echoed on prompt
- b Specifies that the script is to exit when errors occur
- U Specifies the user name
- P Specifies the password
- i Specifies the input file
- o Specifies that all output is to be redirected to a file

Related tasks:

“Manually upgrading the product databases” on page 356
Use this procedure to upgrade product databases manually.

Downtime requirements

Stand-alone and network deployment migrations both require a period of time during which the applications are unavailable.

Stand-alone environments

All three variants of the stand-alone migration procedure result in the stand-alone server being unavailable for the duration of the execution of the procedure.

Network deployment environments

Network deployment migration can be done by following a full downtime procedure or a minimal downtime procedure.

The network full downtime procedure assumes a migration downtime window where the network deployment environment is quiesced, all of the profiles are migrated, the database is upgraded, and the migrated version of the environment is started. The minimal downtime procedure allows half the nodes in a cluster to be migrated while the other half are servicing consumer requests, minimizing downtime to the period where the nodes running the prior version are shut down,

the database is upgraded, and the migrated nodes are started. Use the full downtime procedure if the migration can be completed in the downtime window scheduled for the migration otherwise use the minimal downtime procedure.

What gets migrated

When you use the WebSphere ESB runtime migration procedures to migrate to WebSphere ESB V7.5.1, the following items are migrated: user applications, profile configuration data, adapters, and data sources and providers.

User Applications

Your user applications (any applications not provided with the WebSphere ESB product) are binary-compatible for the supported migration scenarios. When you use the runtime version-to-version migration method, all user applications are automatically migrated to the new target version. You should not have to modify any part of the application to have it run on the newer version of the product. Except for sample applications, applications that are provided as part of the source product are migrated to the latest version of those applications. These are handled as follows:

- **System Applications:** For all system applications—applications that reside in the `install_root/systemApps` directory, the newer version is installed.
- **Support Applications:** For all support applications—applications provided with WebSphere ESB, older versions are updated to the latest version.
- **Sample Applications:** Sample applications are handled differently. For stand-alone profiles, the migration process does not install any sample applications. To make sample applications available for a stand-alone profile, you can install them using the installation wizard for the later version of the product. For network deployment profiles, any samples installed with the previous version of the product will be installed during migration to the new version.

Adapters

For version 6.1.0, 6.1.2 and 6.2.0 WebSphere Adapters, you need to install an interim fix with the name “Mandatory adapter fix for running 6.1 and 6.2 Adapters on WPS v7.0.” Please apply this interim fix on the source environment if you do not plan to update the WebSphere Adapter to a 7.0 level and want to use the application with 6.1.0 or 6.1.2 version of the WebSphere Adapter. For information about how to apply this interim fix, see the appropriate step in Runtime premigration checklist.

The runtime migration procedure for WebSphere ESB version 7.0.0.3 allows you to upgrade the adapter as part of the runtime migration that use previous versions of WebSphere Adapters.

Note: All WebSphere Adapters for version 6.0.2 and Websphere Adapter for SAP versions 6.0.2, 6.1.0, 6.1.2 and 6.2.0 are not supported on WebSphere ESB V7.5.1. This set of adapters has to be updated to V7.5.1 before any applications using them can be deployed on WebSphere ESB V7.5.1. See “Runtime premigration checklist” on page 309 for instructions about how to handle WebSphere Adapters during runtime migration.

Profile configuration data

The version-to-version migration tools (wizard or commands) automatically apply the configuration settings from the previous profile to the new profile created during the migration process.

JDBC providers and data sources

Profile migration automatically migrates the JDBC provider and data source definitions for each existing data source and provider.

What does not get migrated?

Custom files or artifacts do not get automatically migrated. Most of these artifacts are user-created, and are not recognized by WebSphere ESB. Because they are not recognized, they are not migrated.

- **Share-by-reference (shared library) artifacts**

If you are using the share-by-reference pattern for sharing SCA Libraries, then any artifacts that exist in the lib/ext and config directory, such as Java .jar libraries, are not migrated to the migration target. Although, the WebSphere configuration settings for share-by-reference libraries are transferred during profile migration, the actual library .jar artifact should be copied manually post-migration.

- **Most custom files or artifacts added to the WebSphere ESB installation directory or profile directory structure**

Most non-product files, such as custom Jython scripts, are not transferred as part of migration.

Similarly, if you have modified any WebSphere-specific scripts, then these changes need to be manually reapplied to the migration target after migration.

Important: Keep any custom scripts or modified product scripts outside of the installation directory to prevent any accidental deletion of user-modified scripts.

Known compatibility issues

The following items are known compatibility issues when migrating to WebSphere ESB V7.5.1.

SCA Wiring

If you have SCA modules which use a single reference for both dynamic and static invocations, and the reference is wired to an import with a JMS or HTTP binding, then the JMS or HTTP binding will now be used for dynamic invocations using jms: or http: URLs, rather than performing a dynamic web service invocation. To retain the version 6.12 behavior and continue to make Web service invocations in this scenario, you must either update your module to correctly set the bindingType to indicate a web service URL when making the invocation (for MFC or POJO components) or set the WebSphere variable `SCA_USE_WS_FOR_DYNAMIC_INVOCATION` to include the name of the modules in a semi-colon delimited list, for example, `sca/myModule1;sca/myModule2`

Runtime premigration checklist

Before you begin the process of migrating to a new version of WebSphere ESB, you should verify each of the items in this checklist.

- “Hardware, operating system, and database prerequisites”
- “WebSphere ESB installation images”
- “Upgrading DB2 for z/OS to supported versions”
- “Upgrading Oracle database and JDBC driver” on page 310
- “Checking for changesets in the database repository” on page 310
- “Updating applications using WebSphere Adapters, version 6.0.2.x” on page 310
- “Updating applications using CICS Adapters, version 6.x” on page 311
- “Updating WebSphere Adapter for SAP” on page 312
- “Applying the WebSphere Adapter interim fix” on page 311
- “Source profile backup directory storage” on page 313
- “Source database backup storage” on page 313
- “Source profile snapshot directory storage” on page 313
- “Target profile directory storage” on page 313
- “Ulimit setting” on page 314
- “Database authorization” on page 314
- Determining the migration type
- “Migrating root configurations to non-root” on page 314
- “Migrating non-root configurations to root” on page 314
- “Migrating WebSphere Integration Developer” on page 315

Hardware, operating system, and database prerequisites

Verify that your target migration environment is a supported operating environment for WebSphere ESB V7.5.1. This includes the hardware platform, the operating system, and the database.

For hardware and software requirements, visit <http://www-01.ibm.com/software/integration/wsesb/sysreqs/>.

WebSphere ESB installation images

Download the WebSphere ESB installation images and the latest fix packs so they are ready to be installed on each system to be migrated. Validate that there is sufficient storage on the system to have WebSphere ESB and fix packs installed.

Upgrading DB2 for z/OS to supported versions

If your DB2 for z/OS database is at version 7, upgrade it to DB2 for z/OS version 8 or 9 before performing the migration. DB2 for z/OS version 7 is no longer supported by IBM.

After you upgrade your DB2 database to version 8 or 9, you must apply table and index updates to the Common database. Use the SQL in the dbscripts directory to assist you with applying table and index updates.

To access the SQL, go to the following directory:

`/WebSphere/V7T2DM/DeploymentManager/dbscripts/CommonDB/DB2zOSV8`

While in the directory, check for members like `upgradeSchema610DB2z0SV7.sql`, where **610** is the version of WebSphere ESB you are migrating from. If they are present, edit the SQL accordingly and apply the changes to the upgraded WebSphere ESB for z/OS database.

Upgrading Oracle database and JDBC driver

If you are using Oracle 9i, and have not yet upgraded your database to 10g or 11g, download the Oracle 10g or 11g install images and be prepared to upgrade to the new database version as a step in the procedures.

If you are using the Oracle `ojdbc14.jar` or the `ojdbc5.jar` JDBC driver, download the new `ojdbc6.jar` JDBC driver and be prepared to install and configure it as a step in the procedures.

Checking for changesets in the database repository

When migrating from WebSphere ESB V6.2 to WebSphere ESB V7.5.1, check for the existence of changesets in the database repository. A changeset defines one or more changes to be applied to the database.

Perform the following procedure to check for changesets in the database repository:

1. Stop all the servers, either in stand-alone or in a clustered setup.
2. Connect to the CommonDB database schema by using the SQL Command client.
3. Execute the SQL command as follows:

```
select count(*) from w_statement
  where pred_id=(select id from w_uri where uri like '%changeSetState')
    and obj_id IN (select id from w_obj_lit_string where litval IN ('DRAFT', 'PENDING', 'APPROVED'))
    and obj_typ_cd=6
    and version_to=2000000000;
```

If the SQL command reveals database records, there are changesets in the repository that are in the active state.

As a result, no new content will be *bootstrapped* to the repository after migration, resulting in data loss and unexpected behavior. To avoid this scenario, run the following SQL command:

```
update w_statement set obj_id=(select id from w_obj_lit_string where litval='PUBLISHED')
  where pred_id=(select id from w_uri where uri like '%changeSetState')
    and obj_id IN (select id from w_obj_lit_string where litval IN ('DRAFT', 'PENDING', 'APPROVED'))
    and obj_typ_cd=6
    and version_to=2000000000;
```

Updating applications using WebSphere Adapters, version 6.0.2.x

All WebSphere Adapters version 6.0.2.x are not supported by WebSphere ESB version 7.5.1 because of WebSphere Adapters support on runtime for N-3 releases. This lack of compatibility necessitates that you update all the applications that use the WebSphere Adapters version 6.0.2.x and that you update all WebSphere Adapters version 6.0.2.x instances installed at node level to a compatible version for 7.5.1 migration.

Perform the updates using one of the following methods:

- Update the WebSphere Adapters to version 6.1, 6.2, or 7.0 in the source before you attempt runtime migration.

- Uninstall the WebSphere Adapter manually from the source environment before you execute the runtime migration. Update the WebSphere Adapter in IBM Integration Designer and redeploy the adapter to the target environment manually after you have completed runtime migration procedures.

For more information on updating the applications to use WebSphere Adapters V7.5.1, see the WebSphere Adapter documentation on the IBM Integration Designer information center.

Updating applications using CICS Adapters, version 6.x

CICS Adapters version 6.x are not supported during runtime migration. This necessitates that you update all the applications that use the CICS Adapters version 6.x and that you update all CICS Adapters version 6.x instances installed at node level to a compatible version for version 7.5.1.

Perform the updates by uninstalling the CICS Adapter and related applications manually from the source environment before you execute the runtime migration. Update the CICS Adapter in IBM Integration Designer and redeploy the adapter to the target environment manually after you have completed runtime migration procedures.

For more information on updating the applications to use the CICS Adapter V7.5.1, see the WebSphere Adapter documentation on the IBM Integration Designer information center.

Applying the WebSphere Adapter interim fix

If any of the applications in the source environment embed any of the WebSphere Adapters (with the exception of SAP) at version 6.1.0 or version 6.2.0 or use WebSphere Adapter versions 6.1.0 or 6.2.0 configured at the node level, and should not be updated to version 7.5.1 during runtime migration, you must apply an adapter interim fix (iFix) in the source environment before starting the migration procedure. This can be done as follows:

1. Obtain the “Mandatory adapter fix for running 6.1 and 6.2 Adapters on WPS v7.0” for the WebSphere Adapter(s) your applications use. Use one of the following options to obtain the iFix:
 - If you are using WebSphere Adapter version 6.2.x or 6.1.x, contact the IBM support team to obtain the corresponding Adapter iFix.
 - If you are using WebSphere Adapter version 6.1.x, update IBM Integration Designer and extract the RAR file, using the following procedure:
 - a. Download the following version of IBM Integration Designer: WebSphere Integration Developer version 6.1.2 Interim Fix 005.
 - b. Update your existing version of IBM Integration Designer to the new version using Installation Manager.
 - c. Extract the RAR file from the following directory: IBM Integration Designer/*installation_directory* /ResourceAdapters.
2. Apply this fix over the respective WebSphere Adapter on the source environment. Use one of the following procedures, depending on whether the WebSphere Adapter is embedded in the application or installed at the node level.
 - If the WebSphere Adapter is embedded in the application, use the following procedure to apply the iFix:
 - a. Log on to the administrative console.

- b. Select the Application, then click **Update**.
 - c. In the Application update options section, select **Replace or add a single module**, then type the name of the WebSphere Adapter RAR file that represents the single module you want to update.
 - d. Click **Browse** to select the updated RAR file on the local file system that has the changes.
 - e. Select the default values in the remaining steps, then click **Finish**. This will ensure that existing configurations, for example, are not changed and that only JAR files will be updated.
- If the WebSphere Adapter is installed at the node level, use the following procedure to apply the iFix:
 - a. From the administrative console, browse the WebSphere Adapter instance and make note of all Managed Connection Factory and ActivationSpec instances configured for the adapter.
 - b. Select the WebSphere Adapter, then click **Delete** to uninstall the adapter.
 - c. Install the new version of the WebSphere Adapter.
 - d. Configure the Managed Connection Factory and ActivationSpec instance that you made note of in Step a.

Note: If the dependent applications have configuration for ManagedConnectionFactory and ActivationSpec in the .import and .export files, respectively, you can also uninstall and install the application to recreate the configuration for ManagedConnectionFactory and ActivationSpec. If the application uses JNDI reference to configure the ManagedConnectionFactory and ActivationSpec, you must manually recreate the instances as documented in steps 1 and 2.

Updating WebSphere Adapter for SAP

If the source environment is 6.1 or 6.2, you must perform the update activities for the WebSphere Adapter for SAP. If the source environment is 7.0, this is not necessary.

Note: Configure the new SAP application dependency libraries to the target environment *before starting any cluster or server processes that have applications using WebSphere Adapter for SAP*. Without the SAP application libraries configured, the WebSphere Adapter for SAP cannot connect to the SAP application and perform the needed function for the application to work successfully. For details on which libraries to configure and in which directories, see Adding external software dependencies to the server runtime.

6.1.x and 6.2.x versions of WebSphere Adapter for SAP are not supported by WebSphere ESB version 7.5.1 because of incompatible changes introduced by the SAP SAPJCO library updated to support the Java Runtime Environment version 1.6. You must update all the applications that use the WebSphere Adapter for SAP and update all WebSphere Adapter for SAP instances installed at node level to version 7.5.1. You can perform the updates using one of the following methods:

- Update the WebSphere Adapter during runtime migration.
For information on how to update the WebSphere Adapter during runtime migration, see the Runtime migration subprocedures that apply to the version-to-version migration procedure you are using.
- Uninstall the WebSphere Adapter manually from the source environment before executing the runtime migration. Update the WebSphere Adapter in IBM

Integration Designer and redeploy the adapter to the target environment manually after you have completed the runtime migration procedures. For more information on updating the applications to support the new SAP SAPJCO library and the WebSphere Adapter for SAP version 7.5.1, see WebSphere Adapter for SAP documentation.

Source profile backup directory storage

During migration the profile being migrated is backed up in case a rollback is necessary. The space available for the profile backup directory should be at least the size of the configuration directory and applications of the source file.

Source database backup storage

The migration procedures strongly recommend backing up your source product databases before migrating them. Verify that sufficient space exists to back up these databases. The size required for the backups will depend on the size of your production databases and the specifics of your database backup strategy.

Source profile snapshot directory storage

The configuration files in the profile to be migrated are copied during the migration procedure to a snapshot directory that then becomes the source for the profile migration. The directory is an optional parameter for the `BPMSnapshotSourceProfile` command or a configurable value in the WebSphere ESB profile migration wizard and is defaulted to `MigrationSnapshots`.

Before migration, verify that there is sufficient storage for the snapshot directory. The storage requirements can be estimated by summing up the following amounts:

- Size of the profile configuration information to be migrated:
 - `profile_root/installableApps` directory
 - `profile_root/installedApps` directory
 - `profile_root/config` directory
 - `profile_root/properties` directory
- Size of the shared libraries to be migrated:
 - Shared libraries referenced in the `libraries.xml` configuration files
- Size of the resource adapter archives to be migrated:
 - Resource Adapter Archive (RAR) files referenced in the `resources.xml` configuration files
- If trace is enabled, allocate an additional 200 MB (depending on the size and complexity of your configuration) for the trace file written to the snapshot directory.

Target profile directory storage

During migration the target profile is created by using the `BPMCreateTargetProfile` command or the WebSphere ESB profile migration wizard and the source profile is migrated to the target profile referenced from the target installation.

Before migration, verify that there is sufficient storage for the target profile directory. The storage requirements can be estimated by summing up the following amounts:

- Size of the profile configuration information to be migrated:

- *profile_root/installableApps* directory
- *profile_root/installedApps* directory
- *profile_root/config* directory
- *profile_root/properties* directory
- Size of the shared libraries to be migrated:
 - Shared libraries referenced in the *libraries.xml* configuration files
- Size of the resource adapter archives to be migrated:
 - Resource Adapter Archive (RAR) files referenced in the *resources.xml* configuration files
- If trace is enabled, allocate an additional 200 MB (depending on the size and complexity of your configuration) for the trace file written to the snapshot directory.

Ulimit setting

On UNIX systems, to avoid an error during profile migration caused by too many open files, increase the ulimit setting on the system running the profile migration process.

Database authorization

Verify whether you will be able to run all the database scripts using a single user ID, or whether your database administrator might have to run any of them.

See the information in the Databases topic for more information on the required permissions for the product databases.

Determining the migration type

If you are migrating a stand-alone profile, determine whether you plan to do a side-by-side migration, a migration to a remote system, or a migration that requires the operating system on the source system to be upgraded during the migration process.

Migrating root configurations to non-root

If you are migrating a previous version environment with root user permissions to version 7.5.1 with non-root user permissions, complete the steps in the Migrating root configurations to non-root topic on the WebSphere Application Server Version 7.0 information center before attempting the migration procedure.

Note: The reference to *USER_HOME* in the “Migrating root configurations to non-root” instructions refers to the *USER_INSTALL_ROOT* or the root directory of the source profile.

Migrating non-root configurations to root

If you are migrating a previous version environment with non-root user permissions to version 7.5.1 with root user permissions, complete the steps in the Migrating non-root configurations to root topic on the WebSphere Application Server Version 7.0 information center before attempting the migration procedure.

Migrating WebSphere Integration Developer

If you are migrating the target runtime environment to WebSphere ESB version 7.5.1, you will be able to run the applications that you previously deployed, even if you have not migrated WebSphere Integration Developer. However, if you want to edit the applications, you must also migrate WebSphere Integration Developer to a compatible version so the major versions of the two products match.

Related tasks:

“Migrating a network deployment environment with full downtime” on page 322
Use this procedure to migrate a network deployment environment while incurring full downtime.

“Migrating a network deployment environment with minimal downtime” on page 330

Use this procedure to migrate a network deployment environment while incurring minimal downtime.

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

Runtime migration procedures

Use the runtime migration procedures to perform a version-to-version migration.

About runtime migration procedures

The runtime migration tools and documentation support the following three migration procedures: migrating stand-alone environments, migrating network deployment environments with full downtime, and migrating network deployment environments with minimal downtime.

Each of the three runtime migration procedures contains a set of steps and subprocedures. In addition to understanding how the procedures work, it is equally important to consider how you will test the migration procedure you select. The following sections provide an overview of each procedure as well as information to consider about testing the migration.

- “Migrating stand-alone environments”
- “Migrating network deployment environments with full downtime” on page 316
- “Migrating network deployment environments with minimal downtime” on page 316
- “Migration testing” on page 316

Migrating stand-alone environments

The procedure for migrating stand-alone environments describes the steps for backing up the environment, migrating the stand-alone profile, and upgrading the product databases configured for the profile. The procedure contains variants for the different supported mechanisms of migrating a stand-alone environment including side-by-side migration, remote migration, and operating system upgrade migration. Before migrating a stand-alone environment, determine which of these variants fits your requirements best.

For instructions on using this procedure, see “Migrating a stand-alone environment” on page 317.

Migrating network deployment environments with full downtime

There are two different procedures for migrating network deployment environments that differ depending on the length of your downtime migration window. The full downtime procedure is the simplest procedure and is recommended if your downtime window can accommodate the migration. The length of the migration will depend on several factors including the source version, the number of cells, clusters, nodes, applications, and the amount of data in the database. To determine the length your migration will take use the full migration process in your staging environment. It is critical that you follow the network deployment procedure steps carefully and in the order they are listed to ensure that you successfully migrate your network deployment environment.

For instructions on using this procedure, see “Migrating a network deployment environment with full downtime” on page 322.

Migrating network deployment environments with minimal downtime

The minimal downtime procedure should be used if you are unable to accommodate the migration using the full downtime procedure for your migration window and you can accommodate the downtime required for the minimal downtime procedure or in scenarios where the amount of downtime required for the migration directly impacts your business. The minimal downtime procedure is more complex than the full downtime procedure and should only be used when the length of the downtime is critical. If you are not able to accommodate minimal downtime you should consider either the manual or artifact migration methods instead of the runtime migration method. The minimal downtime procedure involves splitting the migration into two groups and migrating one group while the other is still running minimizing the downtime for the cluster. The minimal downtime occurs just prior to bringing the migrated group of nodes online in order to update the database schema and the product data.

Note: If the source version contains applications that exploit Business Calendars or mediation flow components, the minimum downtime procedure cannot be used unless those applications can tolerate some downtime. Nodes with servers that are running applications that exploit Business Calendars or mediation flow components will remain stopped until the node is migrated to version 7.5.1.

For instructions on using this procedure, see “Migrating a network deployment environment with minimal downtime” on page 330.

Migration testing

It is critical that any production migrations are thoroughly tested in a staging environment before they are attempted in a production setting. In addition, it is important that the backup steps in the procedures are followed carefully to enable rollback in cases where the configuration data or the applications failed to migrate successfully to the target environment. Manual and artifact migration methods are often used in conjunction with runtime migration to validate that a typical application or all the applications can be deployed to a version 7.5.1 environment without issue or the applications can be migrated by the development tools successfully thus providing greater assurance that backwards compatibility for the application will be maintained. If you intend to migrate a network deployment environment, it is also helpful to start with a stand-alone environment in a staging environment to learn how to use the tools and the essence of the runtime

migration process before using the more involved network deployment full downtime or minimal downtime procedures.

Migrating a stand-alone environment

Use this procedure to migrate a stand-alone environment.

Before you begin

Review the “Migration overview” on page 289 and BPM runtime premigration checklist topics.

About this task

To migrate a stand-alone environment, use the following procedure.

Procedure

1. Install the migration target products.
 - For a side-by-side migration, install the target product and latest fix packs on the same system as the source product of the migration.
 - For a remote migration, install the target product and latest fix packs on the system that will serve as the target for the migration.
 - For an operating system upgrade migration, defer the installation until after the operating system is upgraded.

Important: You must either install the target version with the same user ID as that used for installing the source version or have permission to access the configuration and data on the source installation.

Important: To migrate from source profiles augmented by multiple products, the new version of those products must be installed into the same target installation directory. For example, if the source profile is augmented by WebSphere ESB and IBM Business Monitor, both of those products must be installed into the same target installation directory.

2. Upgrade DB2 for z/OS and OS/390 Version 7.

If you use DB2 for z/OS and OS/390 Version 7, and have not yet upgraded the database to DB2 for z/OS Version 8 or DB2 9 for z/OS, perform the upgrade now, as described in the DB2 for z/OS documentation.

3. Upgrade Oracle 9i and the Oracle JDBC driver.

Important: You must perform this step on all WebSphere ESB installations that access the Oracle database.

- a. If you are using Oracle 9i and have not yet upgraded your database to 10g or 11g, perform the upgrade now, as described in the Oracle documentation.
- b. If you are using the `ojdbc14.jar` or the `ojdbc5.jar` driver, you must install the new `ojdbc6.jar` driver in the directory that is pointed to by the `ORACLE_JDBC_DRIVER_PATH` WebSphere variable. To do this, use the following procedure.
 - 1) Check the value for the `ORACLE_JDBC_DRIVER_PATH` variable in the previous environment. Use one of the following methods to do this:
 - In the administrative console, select **Environment** > **WebSphere variables**, then select the scope that matches the node of the source profile.

- Navigate to the `variables.xml` file in the following directory:
`source_profile_root\config\cells\cell_name\nodes\node_name\`.

Note: The cell name and node name must match the source profile information.

- 2) Install the new `ojdbc6.jar` driver in the directory that is pointed to by the `ORACLE_JDBC_DRIVER_PATH` WebSphere variable. Use one of the following steps, depending on the location pointed to by the variable.
 - If the variable points to a directory outside of the WebSphere ESB installation, copy the `odbc6.jar` file into the same folder where the `ojdbc14.jar` or `ojdbc5.jar` file resides.
 - If the variable points to a directory within the WebSphere ESB installation, create the same directory structure in the WebSphere ESB V7.5.1 installation, and then copy the `odbc6.jar` file into that directory.
4. Stop the migration source server.

Stop the migration source server using the `stopServer` command from the `profile_root/bin` directory on the migration source system or from the profile's First steps console. Use the following syntax: Linux UNIX

Windows

- Linux UNIX `stopServer.sh server_name -username user_name -password password`
- Windows `stopServer.bat server_name -username user_name -password password`

Note:

- If the profile has security enabled the user name provided must be a member of the operator or administrator role.
- If security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the server.
- If the profile does not have security enabled the `-username` and `-password` parameters are not necessary.

For more information about the `stopServer` command, see the `stopServer` command topic on the WebSphere Application Server, Version 7.0 information center.

5. Back up the migration source profile.

Back up the profile configuration on the migration source server using the `backupConfig` command.

Use the following syntax to back up a profile named `profile1` to

`/ProfileBackups/profile1.zip`. Linux Windows UNIX

- Linux UNIX `backupConfig.sh /ProfileBackups/profile1.zip -profileName profile1`
- Windows `backupConfig.bat c:\ProfileBackups\profile1.zip -profileName profile1`

For more information about the `backupConfig` command, see the `backupConfig` command topic on the WebSphere Application Server, Version 7.0 information center.

6. Back up the .nifRegistry file.

The .nifRegistry file identifies the installation root for all installed WebSphere ESB products; it also identifies the installation root for all installed WebSphere Application Server products. It is located as follows:

UNIX

Linux UNIX /opt/.ibm/.nif/.nifregistry

Windows

- If the user ID that installed the product had administrative privileges, the file is located in the Windows root directory (C:\Windows or C:\WINNT on most Windows systems).
- If the user ID that installed the product did not have administrative privileges, the file is located in the home directory of that user ID.

7. Back up the migration source product databases.

Back up the following databases that are configured by the stand-alone profile according to the documentation for your database:

8. Migrate the stand-alone server profile.

- For a side-by-side migration, the WebSphere ESB profile migration wizard or the WebSphere ESB migration command-line utilities can be used to migrate the source profile.
 - To use the WebSphere ESB profile migration wizard, follow the “Migrating a profile using the profile migration wizard” on page 345 procedure on the system containing the source profile.
 - To use the WebSphere ESB migration command line utilities, follow the Migrating a profile using the BPM migration command-line utilities procedure on the system containing the source profile.
- For a remote migration, follow the Migrating a stand-alone profile to a remote system procedure.
- For an operating system upgrade migration, see the Migrating a stand-alone profile while upgrading an operating system procedure.

9. Upgrade the product databases.

Databases configured for a stand-alone environment are automatically updated as part of starting the server in the following scenarios:

- a. Common database: The database user configured for the Common database has the needed permissions.
- b. Business Process Choreographer: The environment is not using the default table spaces for the Business Process Choreographer database. If the standalone environment is using the sample Business Process Choreographer configuration, or if all of the database objects in the default table spaces specified in the sample SQL scripts have been created, the database uses the default table spaces. This is typically the case for a test environment.

Note: The database user that is configured for the BPEDB data source is not authorized to perform all of the following operations:

- Create and alter tables
- Create and drop indexes and views
- Query, update, delete, and insert (for the table SCHEMA_VERSION)

Note: If you are migrating from version 7.0 or 7.5 to 7.5.1, it is not necessary to perform a schema upgrade for the Common database. Refer to “Migrating databases” on page 356.

Note: Refer to “Databases” on page 303 for the database permissions needed to perform schema upgrades.

The Common Event Infrastructure database and Messaging Engine database are automatically migrated by the runtime migration procedure when the profiles are migrated. For more information, see “Databases” on page 303.

In all other scenarios, the product databases must be upgraded manually. Refer to “Manually upgrading the product databases” on page 356.

If the source version is 7.0.x or 7.5, refer to Step 10 b to migrate the Business Space database schema. Business Space combines the schema upgrade and data migration into one procedure.

Optional: Migrate the messaging engine database if it is needed for your environment. To learn more about when and how to migrate the messaging engine, see the Migrating a messaging engine based on a data store topic on the WebSphere Application Server, Version 7.0 information center.

10. Migrate the instance data for Business Space if it is configured in the source environment. Perform the following steps to update the data in the databases to work with version 7.5.1.
 - a. If the source version is 6.x, migrate the Business Space database data using the procedure described in Migrating the Business Space database data (V6.x). If the source version is 7.0.x or 7.5, migrate the Business Space database schema using the procedure described in Migrating the Business Space V7.0.x or V7.5 database and data.
11. Update the data source configuration. If you have data sources using the embedded data direct driver and you did not update them in the source environment to use a licensed Data Direct JDBC driver or Microsoft JDBC driver, update the Data Source configuration. To do this, use the following procedure.

Attention: The SystemOut.log file might reflect errors because some components could not establish connection to the database.

a. **Start the migration target server.**

Start the migration target server using the startServer command from the *profile_root/bin* directory of the migration target server or from the target profile's First steps console. Use the following syntax:

| | |
|---------|------|
| Windows | UNIX |
| Linux | UNIX |

- `startServer.sh server_name`
- `startServer.bat server_name`

For more information about the startServer command, see the startServer command topic on the WebSphere Application Server, Version 7.0 information center.

- b. Log in to the administrative console.
- c. Update the data source configuration using the following steps.
 - 1) Create a new Data Source with the correct JDBC Provider Type, and set the following properties that match with the existing Data Source: JNDI Name, statementCacheSize, relationalResourceAdapter, authMechanismPreference, authDataAlias, databaseName, serverName, portNumber, and URL.
 - 2) Delete the existing Data Source that uses embedded driver.

3) Use the Test Connection option to check if the data source configuration works.

4) Stop the migration target server.

Stop the migration target server using the `stopServer` command from the `profile_root/bin` directory on the migration source system or from the profile's First steps console. Use the following syntax: Linux

Windows

UNIX

- Linux UNIX `stopServer.sh server_name -username user_name -password password`
- Windows `stopServer.bat server_name -username user_name -password password`

Note:

- If the profile has security enabled, the user name provided must be a member of the operator or administrator role.
- If security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the server.
- If the profile does not have security enabled the `-username` and `-password` parameters are not necessary.

For more information about the `stopServer` command, see the `stopServer` command topic on the WebSphere Application Server, Version 7.0 information center.

12. Start the migration target server.

Start the migration target server using the `startServer` command from the `profile_root/bin` directory of the migration target server or from the target profile's First steps console. Use the following syntax: Linux Windows

UNIX

- Linux UNIX `startServer.sh server_name`
- Windows `startServer.bat server_name`

For more information about the `startServer` command, see the `startServer` command topic on the WebSphere Application Server, Version 7.0 information center.

13. Configure the additional features for WebSphere ESB V7.5.1 (for example, Business Process Definitions), because these are not configured during the runtime migration procedure:

- a. Configure the Process Server components using the procedure described in the topic “Configuring a Process Server”.
- b. Configure the Performance Data Warehouse using the procedure described in the topic “Configuring the Business Performance Data Warehouse component on a server or cluster”.

Results

The stand-alone environment is migrated to the target version.

What to do next

Verify that the migration was successful. For instructions, see “Verifying migration” on page 359.

Related concepts:

“Migration overview” on page 289

The process of moving applications, configuration, and databases from an earlier version of WebSphere ESB to this version of WebSphere ESB is referred to as version-to-version migration, or simply migration.

“Runtime premigration checklist” on page 309

Before you begin the process of migrating to a new version of WebSphere ESB, you should verify each of the items in this checklist.

Related tasks:

“Migrating a profile using the profile migration wizard” on page 345

The profile migration wizard is a graphical user interface (GUI) that guides you through the process of migrating a profile. Migrating a profile is just one step in a series of steps required to migrate a stand-alone or network deployment environment.

“Migrating a profile using the command-line utilities” on page 349

Use this subprocedure for migrating a profile using the command-line utilities.

“Migrating a profile to a remote system” on page 351

Use this procedure for migrating a server profile to a remote system.

“Migrating a server while upgrading an operating system” on page 353

User the following procedure for migrating a server profile on a system whose operating system is being upgraded.

“Manually upgrading the product databases” on page 356

Use this procedure to upgrade product databases manually.

“Verifying migration” on page 359

Verify that your migration was successful by checking the log files and the error files for each migration step, and checking operation with the administrative console.

“Migrating the security configuration for a stand-alone environment” on page 358

You can migrate the security configuration and security settings from a previous version of WebSphere ESB to the current version.

Migrating a network deployment environment with full downtime

Use this procedure to migrate a network deployment environment while incurring full downtime.

Before you begin

Review the “Migration overview” on page 289 and BPM runtime premigration checklist topics.

Procedure

Follow these steps to migrate a network deployment environment while incurring full downtime.

1. Install the migration target product(s).

Install the target product and the latest fix packs on the same system as the source product of the migration.

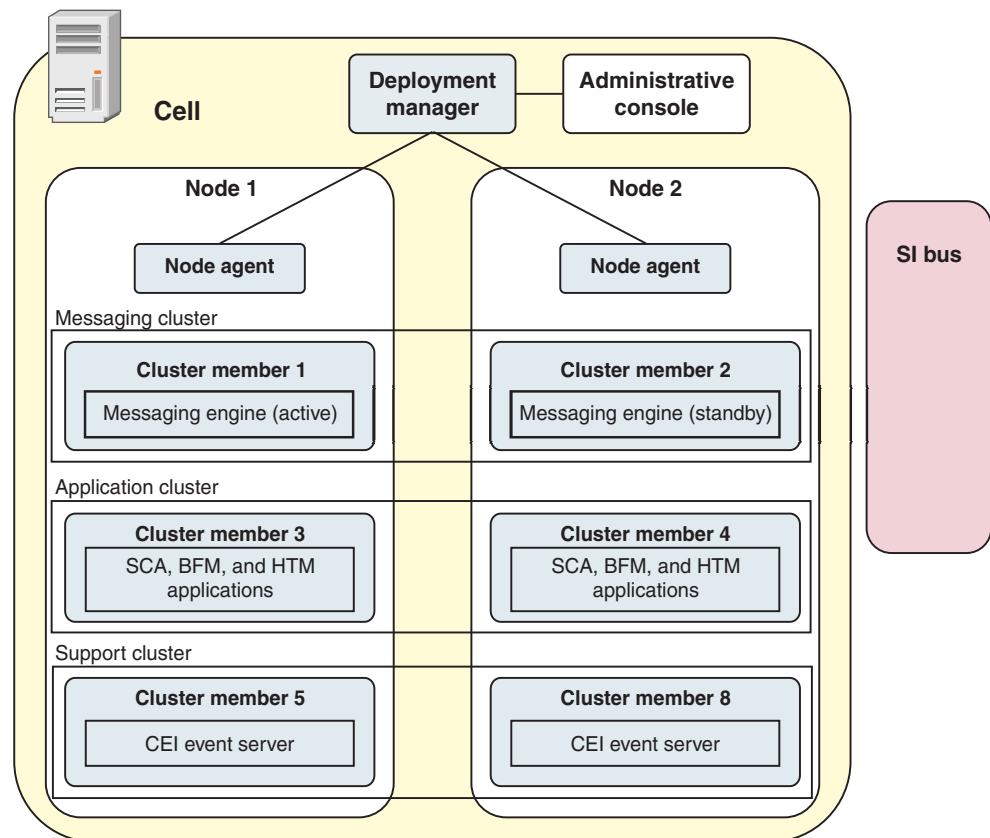
Note: You must either install the target version with the same user ID as that used for installing the source version or have permission to access the configuration and data on the source installation.

Note: To migrate from source profiles augmented by multiple products, the new version of those products must be installed into the same target installation directory. For example, if the source profile is augmented by WebSphere ESB and IBM Business Monitor, both of those products must be installed into the same target installation directory.

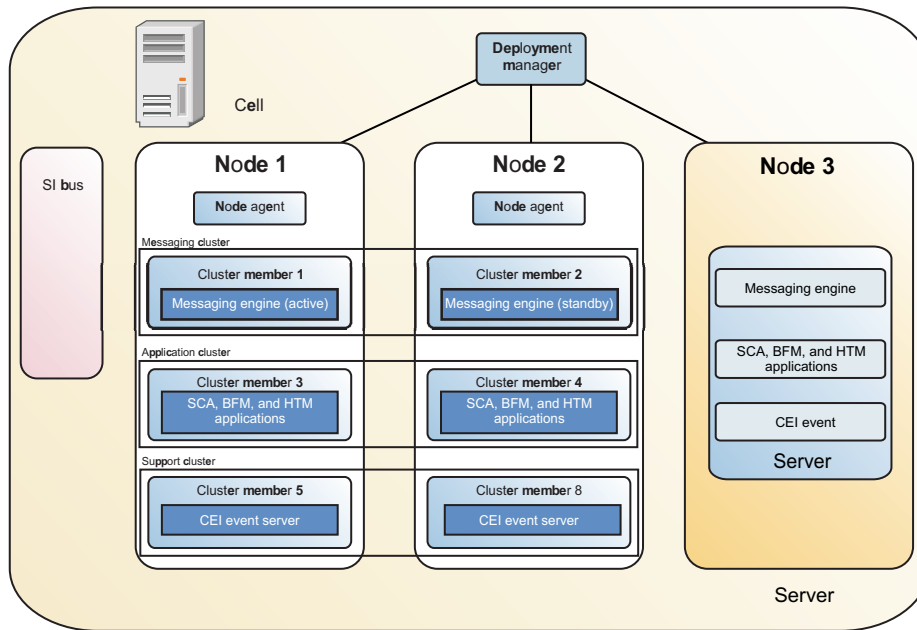
2. Upgrade DB2 for z/OS and OS/390 Version 7.

If you use DB2 for z/OS and OS/390 Version 7, and have not yet upgraded the database to DB2 for z/OS Version 8 or DB2 9 for z/OS, perform the upgrade now, as described in the DB2 for z/OS documentation.

3. Identify the Java Virtual Machines (JVMs) that must be stopped. The following diagram illustrates the logical components of a remote messaging and remote support topology pattern for a network deployment configuration (golden topology).



The following diagram illustrates the logical components of a remote messaging and remote support topology pattern for a network deployment configuration with an additional node that hosts a non-clustered server.



4. Stop the deployment manager, node agents, application server clusters, and non-clustered servers.

- a. Stop the deployment manager.

Stop the source deployment manager using the `stopManager` command from the `profile_root/bin` directory on the migration source system or from the profile's First steps console.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `stopManager.sh -username user_name -password password`
- Windows `stopManager.bat -username user_name -password password`

If the profile has security enabled, the user name provided must be a member of the operator or administrator role.

If security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the server.

If the profile does not have security enabled, the `-username` and `-password` parameters are unnecessary.

For more information about the `stopManager` command, see the `stopManager` command topic on the WebSphere Application Server, Version 7.0 information center.

- b. Stop the node agents.
 - c. Stop the application server clusters.
 - d. Stop the non-clustered servers.
5. Back up the migration source profiles, the `.nifRegistry` file, and the migration source product databases.
 - a. Back up the migration source profiles.

Repeat this step for each profile that will be migrated, including the deployment manager, each non-clustered managed node, and each managed node.

Back up the profile configuration on the migration source server using the backupConfig command.

Use the following syntax to back up a profile named profile1 to

/ProfileBackups/profile1.zip. Linux UNIX Windows

- Linux UNIX backupConfig.sh /ProfileBackups/profile1.zip
-profileName profile1
- Windows backupConfig.bat c:\ProfileBackups\profile1.zip
-profileName profile1

For more information about the backupConfig command, see the backupConfig command topic on the WebSphere Application Server, Version 7.0 information center.

b. Back up the .nifRegistry file.

The .nifRegistry file identifies the installation root for all installed IBM Business Process Manager products; it also identifies the installation root for all installed WebSphere Application Server products. It is located as follows: Linux Windows UNIX

- Linux UNIX /opt/.ibm/.nif/.nifregistry
- Windows
 - If the user ID that installed the product had administrative privileges, the file is located in the Windows root directory (C:\Windows or C:\WINNT on most Windows systems).
 - If the user ID that installed the product did not have administrative privileges, the file is located in the home directory of that user ID.

c. Back up the migration source product databases.

Back up the following databases that are configured by any of the migration source profiles according to the documentation for your database:

- Business Space database
- Common database
- Common Event Infrastructure Database
- Messaging Engine Database

6. Migrate the deployment manager profile.

- For a side-by-side migration, you can use the WebSphere ESB profile migration wizard or the WebSphere ESB migration command-line utilities to migrate the source profile.
 - To use the WebSphere ESB profile migration wizard, follow the Migrating a profile using the BPM profile migration wizard procedure on the system containing the source profile.
 - To use the WebSphere ESB migration command-line utilities, follow the Migrating a profile using the BPM migration command-line utilities procedure on the system containing the source profile.
- For a remote migration, follow the “Migrating a profile to a remote system” on page 351 procedure.
- For an operating system upgrade migration, follow the “Migrating a server while upgrading an operating system” on page 353 procedure.

Note: If you are migrating from IBM Business Process Manager V6.0.2, you must use the Migrating a profile using the BPM migration command-line utilities procedure.

7. Upgrade the product databases.

Databases configured for a stand-alone environment are automatically updated as part of starting the server in the following scenarios:

- a. Common database: The database user configured for the Common database has the needed permissions.
- b. Business Process Choreographer: The environment is not using the default table spaces for the Business Process Choreographer database. If the standalone environment is using the sample Business Process Choreographer configuration, or if all of the database objects in the default table spaces specified in the sample SQL scripts have been created, the database uses the default table spaces. This is typically the case for a test environment.

Note: The database user that is configured for the BPEDB data source is not authorized to perform all of the following operations:

- Create and alter tables
- Create and drop indexes and views
- Query, update, delete, and insert (for the table SCHEMA_VERSION)

Important: Upgrade only the databases for Business Process Choreographer and Business Space that are configured at the server scope under the non-clustered node that is being migrated.

Note: If you are migrating from version 7.0 or 7.5 to 7.5.1, it is not necessary to perform a schema upgrade for the Common database. Refer to “Migrating databases” on page 356.

Note: Refer to “Databases” on page 303 for the database permissions needed to perform schema upgrades.

In all other scenarios, the product databases must be upgraded manually. Refer to “Manually upgrading the product databases” on page 356.

If the source version is 7.0.x or 7.5, refer to step 14 on page 329 b to migrate the Business Space database schema. Business Space combines the schema upgrade and data migration into one procedure.

The Common Event Infrastructure database and Messaging Engine database are automatically migrated when the server process is started post migration. For more information, see “Databases” on page 303.

Optional: Migrate the messaging engine database if it is needed for your environment. To learn more about when and how to migrate the messaging engine, see the Migrating a messaging engine based on a data store topic on the WebSphere Application Server, Version 7.0 information center.

Important: Because the deployment manager uses the Common database, you must complete the Common database upgrade manually, if it cannot be updated automatically, before moving to Step 8. The remaining databases for Business Process Choreographer and Business Space can be updated in parallel while the managed profiles and clusters are being migrated under Steps 10 on page 327 and 11 on page 328.

Important: Make sure that Step 7 has been completed for all databases configured for Business Process Choreographer and Business Space before starting the target servers.

8. Start the target deployment manager.

Start the target deployment manager using the `startManager` command from the *profile_root/bin* directory on the deployment manager system or from the First steps console for the deployment manager profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startManager.sh`
- Windows `startManager.bat`

For more information about the `startManager` command, see the `startManager` command topic on the WebSphere Application Server, Version 7.0 information center.

9. If applicable, update the data source configuration. If you have data sources using a Microsoft JDBC driver, update the Data Source configuration. To do this, use the following procedure.

Attention: The `SystemOut.log` file might reflect errors because some components could not establish connection to the database.

10. Perform the following actions for all nodes that are federated under the deployment manager.
 - a. Migrate the managed nodes.
 - For a side-by-side migration, you can use the WebSphere ESB profile migration wizard or the WebSphere ESB migration command-line utilities to migrate the source profile.
 - To use the WebSphere ESB profile migration wizard, follow the Migrating a profile using the BPM profile migration wizard procedure on the system containing the source profile.
 - To use the WebSphere ESB migration command-line utilities, follow the Migrating a profile using the BPM migration command-line utilities procedure on the system containing the source profile.
 - For a remote migration, follow the “Migrating a profile to a remote system” on page 351 procedure.
 - For an operating system upgrade migration, follow the “Migrating a server while upgrading an operating system” on page 353 procedure.

Note: If you are migrating from IBM Business Process Manager V6.0.2, you must use the Migrating a profile using the BPM migration command-line utilities procedure.

- b. Optional: Migrate the Business Process Rules Manager.

The Business Process Rules Manager is automatically migrated when the last node in the cell is migrated, but if the migrated cluster contains the Business Process Rules Manager it can manually be migrated.

To manually migrate the Business Process Rules Manager for the cluster `cluster1` use the following command:

```
wsadmin -f installBRManager.jacl -cl cluster1
```

For more information about the `installBRManager` command, see the `installBRManager` command-line utility topic.

- c. Start the migration target node agents.

Repeat this step for each non-clustered managed node that was migrated, and each clustered managed node for each cluster that was migrated.

Start the migration target node agent using the `startNode` command from the *profile_root/bin* directory of the migration target server or from the profile's First steps console.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startNode.sh`
- Windows `startNode.bat`

For more information about the `startNode` command, see the `startNode` command topic on the WebSphere Application Server, Version 7.0 information center.

11. Migrate the cluster configuration for clustered nodes.

a. Migrate the cluster-scoped configuration.

Migrate the cluster-scoped configuration using the `BPMigrateCluster` command from the `install_root/bin` directory on the system containing the deployment manager.

Use the following syntax to migrate a cluster named `applicationCluster1` with a deployment manager profile named `dmgrProfile` copied in the `/MigrationSnapshots/ProcServer` directory: Linux UNIX Windows

- Linux UNIX `BPMigrateCluster.sh /MigrationSnapshots/ProcServer applicationCluster1 dmgrProfile`
- Windows `BPMigrateCluster.bat c:\MigrationSnapshots\ProcServer applicationCluster1 dmgrProfile`

For more information about the `BPMigrateCluster` command, see the reference section.

12. Configure the additional features for WebSphere ESB V7.5.1 (for example, Business Process Definitions), because these are not configured during the runtime migration procedure:

- Configure the Process Server components using the procedure described in the topic “Configuring a Process Server”.
- Configure the Performance Data Warehouse using the procedure described in the topic “Configuring the Business Performance Data Warehouse component on a server or cluster”.

13. Synchronize all the clustered nodes that participate in the clusters migrated in previous steps to update the cluster configuration.

- Stop the migration source's node agent using the `stopNode` command from the `profile_root/bin` directory of the migration source system. Use the following syntax: Linux Windows UNIX

- Linux UNIX `stopNode.sh -username user_name -password password`
- Windows `stopNode.bat -username user_name -password password`

- Back up your managed profiles. This backup is needed in case the next step for running the `syncNode` command fails. After resolving the issue with `syncNode`, you can restore the backup before running the `syncNode` command again.

Back up the profile configuration on the clustered managed node using the `backupConfig` command. Use the following syntax to back up a profile named `profile1` to `/ProfileBackups/profile1.zip`: Linux Windows

- UNIX
- Linux UNIX `backupConfig.sh /ProfileBackups/profile1.zip -profileName profile1`

- **Windows** backupConfig.bat c:\ProfileBackups\profile1.zip
-profileName profile1

For more information about the **backupConfig** command, see the backupConfig command topic on the WebSphere Application Server, Version 7.0 information center.

- c. Synchronize the managed nodes. Synchronize the node with the target deployment manager using the **syncNode** command from the profile_root/bin directory of the migration target profile or from the target profile's First steps console. Use the following syntax: **Linux**

Windows **UNIX**

- **Linux** **UNIX** syncNode.sh
deployment_manager_machine_name_or_ip_address
deployment_manager_port_no
- **Windows** syncNode.bat
deployment_manager_machine_name_or_ip_address
deployment_manager_port_no

For more information about the **syncNode** command, see the syncNode command topic on the WebSphere Application Server, Version 7.0 information center.

14. Migrate the instance data for Business Space if they are configured in the source environment. Perform the following steps to update the data in the databases to work with version 7.5.1.
 - a. If the source version is 6.x, migrate the Business Space database data using the procedure described in Migrating the Business Space database data (V6.x). If the source version is 7.0.x or 7.5, migrate the Business Space database schema using the procedure described in Migrating the Business Space V7.0.x or V7.5 database and data.

15. Start the migration target servers.

Repeat this step for each server configured for each non-clustered managed node that was migrated and for each clustered managed node that was migrated.

Start the migration target server using the startServer command from the profile_root/bin directory of the migration target server or from the profile's First steps console.

Use the following syntax: **Linux** **Windows** **UNIX**

- **Linux** **UNIX** startServer.sh server_name
- **Windows** startServer.bat server_name

For more information about the startServer command, see the startServer command topic on the WebSphere Application Server, Version 7.0 information center.

16. Optional: Uninstall the source deployment manager.

Once the migration is complete, the migration source deployment manager can be uninstalled.

17. Remove Compatibility Mode.

If you chose the compatibility option (which is the default), and if all your nodes are completely migrated to the target version, run the convertScriptCompatibility script from the install_root/bin directory on the deployment to remove compatibility.

Use the following syntax: `Linux` `UNIX` `Windows`

- `Linux` `UNIX` `convertScriptCompatibility.sh`
- `Windows` `convertScriptCompatibility.bat`

For more information about the `convertScriptCompatibility` command, see the `convertScriptCompatibility` command topic on the WebSphere Application Server, Version 7.0 information center.

Results

The network deployment environment is migrated to the target version.

What to do next

Verify that the migration was successful. For instructions, see “Verifying migration” on page 359.

Related concepts:

“Migration overview” on page 289

The process of moving applications, configuration, and databases from an earlier version of WebSphere ESB to this version of WebSphere ESB is referred to as version-to-version migration, or simply migration.

“Runtime premigration checklist” on page 309

Before you begin the process of migrating to a new version of WebSphere ESB, you should verify each of the items in this checklist.

Related tasks:

“Migrating a profile using the profile migration wizard” on page 345

The profile migration wizard is a graphical user interface (GUI) that guides you through the process of migrating a profile. Migrating a profile is just one step in a series of steps required to migrate a stand-alone or network deployment environment.

“Migrating a profile using the command-line utilities” on page 349

Use this subprocedure for migrating a profile using the command-line utilities.

“Manually upgrading the product databases” on page 356

Use this procedure to upgrade product databases manually.

“Verifying migration” on page 359

Verify that your migration was successful by checking the log files and the error files for each migration step, and checking operation with the administrative console.

Related reference:

topics/rmig_vtv_bpmmmigratecluster.dita

The **BPMmigrateCluster** command migrates cluster-scoped application and configuration information.

 `installBRManager` command-line utility

The **installBRManager** command migrates the Business Rules Manager.

Migrating a network deployment environment with minimal downtime

Use this procedure to migrate a network deployment environment while incurring minimal downtime.

Before you begin

Review the “Migration overview” on page 289 and “Runtime premigration checklist” on page 309 topics.

Note: If the source version contains applications that exploit Business Calendars or mediation flow components, the minimum downtime procedure cannot be used unless those applications can tolerate some downtime. Nodes with servers that are running applications that exploit Business Calendars or mediation flow components will remain stopped until the node is migrated to version 7.5.1.

Procedure

Follow these steps to migrate a network deployment environment while incurring minimal downtime.

1. Install the migration target product(s).

Install the target product and the latest fix packs on the same system as the source product of the migration.

Note: You must either install the target version with the same user ID as that used for installing the source version or have permission to access the configuration and data on the source installation.

Note: To migrate from source profiles augmented by multiple products, the new version of those products must be installed into the same target installation directory. For example, if the source profile is augmented by WebSphere ESB and IBM Business Monitor, both of those products must be installed into the same target installation directory.

2. Upgrade DB2 for z/OS and OS/390 Version 7.

If you use DB2 for z/OS and OS/390 Version 7, and have not yet upgraded the database to DB2 for z/OS Version 8 or DB2 9 for z/OS, perform the upgrade now, as described in the DB2 for z/OS documentation.

3. Identify the clusters, clustered managed nodes, and non-clustered managed nodes to migrate.

If you intend to migrate an entire cell, you will migrate:

- The deployment manager
- All nodes that do not have an application server that is a member of any cluster in the cell (non-clustered managed nodes)
- All clusters and all the nodes that have application servers that are members of those clusters (clustered managed nodes)

If you are **not** migrating an entire cell, you **do not** intend to migrate any clusters, and you **do** intend to migrate one or more nodes that do not have an application server that is a member of any cluster in the cell (non-clustered managed nodes), you will migrate:

- The deployment manager
- Each non-clustered managed node you intend to migrate

If you are **not** migrating an entire cell, intend to migrate one or more clusters in the cell, and zero or more non-clustered managed nodes, you will migrate:

- The deployment manager.
- Each non-clustered managed node you intend to migrate.

- Each cluster you explicitly intend to migrate and all the nodes that have an application server that is a member of that cluster (clustered managed nodes).
- Any cluster and that cluster's clustered managed nodes implicitly impacted by the clusters that you intend to migrate. To identify the transitive closure of all the impacted clusters and their clustered managed nodes, use the following procedure:
 - For each cluster you intend to migrate, identify all the clustered managed nodes that have application servers that contribute to the cluster.
 - For each clustered managed node, determine what other clusters, if any, the application servers running on the node are members of.
 - Repeat the process for each of these clusters to determine the complete set of clusters and clustered managed nodes that must be migrated as part of this procedure.

4. Disable synchronization for all nodes.

Disable synchronization for all non-clustered managed nodes and clustered managed nodes using the administrative console on the source deployment manager.

- a. From the WebSphere Application Server administrative console, select **System administration > Node agents**.
- b. Click the node agent for the node.
- c. Click **File synchronization service**.
- d. Make a note of the following settings so they can be restored later in the procedure when you re-enable node synchronization:
 - **Enable service at server startup**
 - **Automatic synchronization**
 - **Startup synchronization**
- e. Unselect the following options:
 - **Enable service at server startup**
 - **Automatic migration**
 - **Startup synchronization**
- f. Click **Apply**, then click **OK** to save the configuration changes and to make sure that all nodes in the cell are synchronized to make the changes effective on the node agents.

5. Stop the deployment manager.

Stop the migration source's deployment manager using the `stopManager` command from the *profile_root/bin* directory on the migration source system or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `stopManager.sh -username user_name -password password`
- Windows `stopManager.bat -username user_name -password password`

If the profile has security enabled, the user name provided must be a member of the operator or administrator role.

If security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the server.

If the profile does not have security enabled, the `-username` and `-password` parameters are unnecessary.

For more information about the `stopManager` command, see the `stopManager` command topic on the WebSphere Application Server, Version 7.0 information center.

6. Back up the source deployment manager profile.

Back up the deployment manager profile configuration on the source deployment manager system using the **backupConfig** command.

Use the following syntax to back up a profile named `dmgrProfile` to `/ProfileBackups/dmgrProfile.zip`.

- **Linux** **UNIX** `backupConfig.sh /ProfileBackups/dmgrProfile.zip -profileName dmgrProfile`
- **Windows** `backupConfig.bat c:\ProfileBackups\profile1.zip -profileName dmgrProfile`

For more information about the `backupConfig` command, see the `backupConfig` command topic on the WebSphere Application Server, Version 7.0 information center.

7. Back up the `.nifRegistry` file.

The `.nifRegistry` file identifies the installation root for all installed WebSphere ESB products; it also identifies the installation root for all installed WebSphere Application Server products. It is located as follows:

UNIX

- **Linux** **UNIX** `/opt/.ibm/.nif/.nifregistry`

Windows

- If the user ID that installed the product had administrative privileges, the file is located in the Windows root directory (`C:\Windows` or `C:\WINNT` on most Windows systems).
- If the user ID that installed the product did not have administrative privileges, the file is located in the home directory of that user ID.

8. Back up the cell-scoped Common database.

Back up the cell-scoped Common database using the documentation for your database server.

9. Migrate the deployment manager profile.

- For a side-by-side migration, you can use the WebSphere ESB profile migration wizard or the WebSphere ESB migration command-line utilities to migrate the source profile.
 - To use the WebSphere ESB profile migration wizard, follow the Migrating a profile using the BPM profile migration wizard procedure on the system containing the source profile.
 - To use the WebSphere ESB migration command-line utilities, follow the Migrating a profile using the BPM migration command-line utilities procedure on the system containing the source profile.
- For a remote migration, follow the “Migrating a profile to a remote system” on page 351 procedure.
- For an operating system upgrade migration, follow the “Migrating a server while upgrading an operating system” on page 353 procedure.

Note: If you are migrating from IBM Business Process Manager V6.0.2, you must use the Migrating a profile using the BPM migration command-line utilities procedure.

10. Upgrade the cell-scoped Common database.

If the database user that is defined for the Common database data source does not have sufficient privileges, upgrade the Common database schema manually using the “Manually upgrading the product databases” on page 356 procedure.

Important: Do not upgrade the databases for Business Process Choreographer or Business Space, because they are configured at a cluster or server scope and should be updated only when the respective managed nodes have been migrated to version 7.5.1, which happens later in Steps 18 on page 336 and 36 on page 341.

Note: If you are migrating from version 7.0 or 7.5 to 7.5.1, it is not necessary to perform a schema upgrade for the Common database. Refer to “Migrating databases” on page 356.

Note: Refer to “Databases” on page 303 for the database permissions needed to perform schema upgrades.

11. Start the target deployment manager.

Start the target deployment manager using the `startManager` command from the `profile_root/bin` directory on the deployment manager system or from the First steps console for the deployment manager profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startManager.sh`
- Windows `startManager.bat`

For more information about the `startManager` command, see the `startManager` command topic on the WebSphere Application Server, Version 7.0 information center.

12. Migrate the non-clustered managed nodes.

Repeat steps 12 through 23 on page 337 for each non-clustered managed node that is a source of the migration.

13. Stop the non-clustered managed node migration source servers.

Stop the migration source server using the `stopServer` command from the `profile_root/bin` directory on the migration source system or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `stopServer.sh server_name -username user_name -password password`
- Windows `stopServer.bat server_name -username user_name -password password`

If the profile has security enabled, the user name provided must be a member of the operator or administrator role.

If security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the server.

If the profile does not have security enabled, the `-username` and `-password` parameters are unnecessary.

For more information about the `stopServer` command, see the `stopServer` command topic on the WebSphere Application Server, Version 7.0 information center.

14. Stop the non-clustered managed node migration source node agent.

Stop the migration source's node agent using the `stopNode` command from the `profile_root/bin` directory of the migration source system.

Use the following syntax: `Linux` `UNIX` `Windows`

- `Linux` `UNIX` `stopNode.sh -username user_name -password password`
- `Windows` `stopNode.bat -username user_name -password password`

If the profile has security enabled, the user name provided must be a member of the operator or administrator role.

If security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the system.

If the profile does not have security enabled, the `-username` and `-password` parameters are unnecessary.

For more information about the `stopNode` command, see the `stopNode` command topic on the WebSphere Application Server, Version 7.0 information center.

15. Back up the non-clustered managed node migration source profile.

Back up the profile configuration on the non-clustered managed node using the `backupConfig` command.

Use the following syntax to back up a profile named `profile1` to `/ProfileBackups/profile1.zip`. `Linux` `UNIX` `Windows`

- `Linux` `UNIX` `backupConfig.sh /ProfileBackups/profile1.zip -profileName profile1`
- `Windows` `backupConfig.bat c:\ProfileBackups\profile1.zip -profileName profile1`

For more information about the `backupConfig` command, see the `backupConfig` command topic on the WebSphere Application Server, Version 7.0 information center.

16. Back up the server-scoped product databases configured for the non-clustered managed node.

Back up the following product databases that are configured for the non-clustered managed node according to the documentation for your database:

17. Migrate the non-clustered managed node.

Use the WebSphere ESB profile migration wizard or the WebSphere ESB migration command-line utilities to migrate the non-clustered managed node source profile.

- To use the WebSphere ESB profile migration wizard, follow the Migrating a profile using the BPM profile migration wizard procedure on the system containing the non-clustered managed node profile.

- To use the WebSphere ESB migration command-line utilities, follow the Migrating a profile using the BPM migration command-line utilities procedure on the system containing the non-clustered managed node profile.

Note: If you are migrating from IBM Business Process Manager V6.0.2, you must use the Migrating a profile using the BPM migration command-line utilities procedure.

18. Upgrade the product databases.

Databases configured for the network deployment environment are automatically updated as part of starting the server in the following scenario:

- a. Business Process Choreographer: The environment is not using the default table spaces for the Business Process Choreographer database. If the standalone environment is using the sample Business Process Choreographer configuration, or if all of the database objects in the default table spaces specified in the sample SQL scripts have been created, the database uses the default table spaces. This is typically the case for a test environment.

Note: The database user that is configured for the BPEDB data source is not authorized to perform all of the following operations:

- Create and alter tables
- Create and drop indexes and views
- Query, update, delete, and insert (for the table SCHEMA_VERSION)

Important: Upgrade only the databases for Business Process Choreographer and Business Space that are configured at the server scope under the non-clustered node that is being migrated.

Note: If you are migrating from version 7.0 or 7.5 to 7.5.1, it is not necessary to perform a schema upgrade for the Common database. Refer to “Migrating databases” on page 356.

Note: Refer to “Databases” on page 303 for the database permissions needed to perform schema upgrades.

The Common Event Infrastructure database and Messaging Engine database are automatically migrated by the runtime migration procedure when the profiles are migrated. For more information, see “Databases” on page 303.

In all other scenarios, the product databases must be upgraded manually. Refer to “Manually upgrading the product databases” on page 356.

If the source version is 7.0.x or 7.5, refer to Step 19 b to migrate the Business Space database schema. Business Space combines the schema upgrade and data migration into one procedure.

Optional: Migrate the messaging engine database if it is needed for your environment. To learn more about when and how to migrate the messaging engine, see the Migrating a messaging engine based on a data store topic on the WebSphere Application Server, Version 7.0 information center.

19. Migrate the instance data for Business Space if they are configured in the source environment. Perform the following steps to update the data in the databases to work with version 7.5.1.
 - a. If the source version is 6.x, migrate the Business Space database data using the procedure described in Migrating the Business Space database data (V6.x). If the source version is 7.0.x or 7.5, migrate the Business Space

database schema using the procedure described in Migrating the Business Space V7.0.x or V7.5 database and data.

20. Optional: Migrate the Business Process Rules Manager.

The Business Process Rules Manager is automatically migrated when the last node in the cell is migrated but if the migrated non-clustered managed node contains the Business Process Rules Manager it can manually be migrated.

To manually migrate the Business Process Rules Manager for the server `server1` and the non-clustered managed node `node1` use the following command:

```
wsadmin -f installBRManager.jacl -s server1 -n node1
```

For more information about the `installBRManager` command, see the `installBRManager` command topic.

21. Enable synchronization for the non-clustered managed node.

Enable synchronization for the non-clustered managed node that has been migrated using administrative console on the target deployment manager.

- a. From the WebSphere Application Server administrative console, select **System administration > Node agents**.
- b. Click the node agent for the node.
- c. Click **File synchronization service**.
- d. Restore the settings for:
 - **Enable service at server startup**
 - **Automatic synchronization**
 - **Startup synchronization**
- e. Click **Apply**, then click **OK** to save the configuration changes and to make sure that all nodes in the cell are synchronized to make the changes effective on the node agents.

22. Start the migration target non-clustered managed node agent.

Start the node agent of the migration target non-clustered managed node using the `startNode` command from the `profile_root/bin` directory of the migration target server or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startNode.sh`
- Windows `startNode.bat`

For more information about the `startNode` command, see the `startNode` command topic on the WebSphere Application Server, Version 7.0 information center.

23. Start the migration target non-clustered managed node server.

Start the migration target non-clustered managed node target server using the `startServer` command from the `profile_root/bin` directory of the migration target server or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startServer.sh server_name`
- Windows `startServer.bat server_name`

For more information about the `startServer` command, see the `startServer` command topic on the WebSphere Application Server, Version 7.0 information center.

24. Migrate the clusters.

Repeat 24 on page 337 through 43 on page 343 for each cluster in the network deployment environment that needs to be migrated.

Divide the nodes that contain servers that contribute to the cluster into two roughly equivalent sized groups, group A and group B. The group B nodes will continue to service consumer requests while the group A nodes are taken offline and migrated. When the group A nodes are migrated, all nodes will be stopped, the databases configured for the cluster will be migrated, and the migrated group A nodes will be started and can begin servicing consumer requests. The group B nodes will then be migrated and started. Staggering the migration over the two groups of nodes will minimize the amount of time the cluster will need to be down in order to migrate the product databases.

25. Stop the group A clustered managed node migration source servers.

Repeat this step for each server associated with a clustered managed node that will be migrated as part of group A.

Stop the migration source server using the stopServer command from the *profile_root/bin* directory on the migration source system or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX stopServer.sh server_name -username user_name -password password
- Windows stopServer.bat server_name -username user_name -password password

If the profile has security enabled, the user name provided must be a member of the operator or administrator role.

If security is enabled, the -username and -password parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the server.

If the profile does not have security enabled, the -username and -password parameters are unnecessary.

For more information about the stopServer command, see the stopServer command topic on the WebSphere Application Server, Version 7.0 information center.

26. Stop the group A clustered managed node migration source node agents.

Repeat this step for each node agent associated with a clustered managed node that will be migrated as part of group A.

Repeat this step for each node agent that is impacted by the migration.

Stop the migration source's node agent using the stopNode command from the *profile_root/bin* directory of the migration source system.

Use the following syntax: Linux UNIX Windows

- Linux UNIX stopNode.sh -username user_name -password password
- Windows stopNode.bat -username user_name -password password

If the profile has security enabled, the user name provided must be a member of the operator or administrator role.

If security is enabled, the -username and -password parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the system.

If the profile does not have security enabled, the `-username` and `-password` parameters are unnecessary.

For more information about the `stopNode` command, see the `stopNode` command topic on the WebSphere Application Server, Version 7.0 information center.

27. Back up the group A migration source profiles.

Repeat this step for each profile that will be migrated in group A.

Back up the profile configuration on the non-clustered managed node using the `backupConfig` command.

Use the following syntax to back up a profile named `profile1` to

`/ProfileBackups/profile1.zip`. Linux UNIX Windows

- Linux UNIX `backupConfig.sh /ProfileBackups/profile1.zip -profileName profile1`
- Windows `backupConfig.bat c:\ProfileBackups\profile1.zip -profileName profile1`

For more information about the `backupConfig` command, see the `backupConfig` command topic on the WebSphere Application Server, Version 7.0 information center.

28. Migrate the group A managed nodes.

This step should be repeated for each group A managed node in the cluster.

- For a side-by-side migration, you can use the WebSphere ESB profile migration wizard or the WebSphere ESB migration command-line utilities to migrate the source profile.
 - To use the WebSphere ESB profile migration wizard, follow the Migrating a profile using the BPM profile migration wizard procedure on the system containing the source profile.
 - To use the WebSphere ESB migration command-line utilities, follow the Migrating a profile using the BPM migration command-line utilities procedure on the system containing the source profile.
- For a remote migration, follow the “Migrating a profile to a remote system” on page 351 procedure.
- or an operating system upgrade migration, follow the “Migrating a server while upgrading an operating system” on page 353 procedure.

Note: If you are migrating from IBM Business Process Manager V6.0.2, you must use the Migrating a profile using the BPM migration command-line utilities procedure.

29. Stop the group B clustered managed node migration source servers.

Repeat this step for each server associated with a clustered managed node that will be migrated as part of group B.

Stop the migration source server using the `stopServer` command from the `profile_root/bin` directory on the migration source system or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `stopServer.sh server_name -username user_name -password password`
- Windows `stopServer.bat server_name -username user_name -password password`

If the profile has security enabled, the user name provided must be a member of the operator or administrator role.

If security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the server.

If the profile does not have security enabled, the `-username` and `-password` parameters are unnecessary.

For more information about the `stopServer` command, see the `stopServer` command topic on the WebSphere Application Server, Version 7.0 information center.

30. Stop the group B clustered managed node migration source node agents.

Repeat this step for each node agent associated with a clustered managed node that will be migrated as part of group B.

Repeat this step for each node agent that is impacted by the migration.

Stop the migration source's node agent using the `stopNode` command from the `profile_root/bin` directory of the migration source system.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `stopNode.sh -username user_name -password password`
- Windows `stopNode.bat -username user_name -password password`

If the profile has security enabled, the user name provided must be a member of the operator or administrator role.

If security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the system.

If the profile does not have security enabled, the `-username` and `-password` parameters are unnecessary.

For more information about the `stopNode` command, see the `stopNode` command topic on the WebSphere Application Server, Version 7.0 information center.

31. Migrate the cluster.

Migrate the cluster-scoped profile using the `BPMigrateCluster` command from the `install_root/bin` directory on the system containing the deployment manager.

Use the following syntax to migrate a cluster named `applicationCluster1` with a deployment manager profile named `dmgrProfile` copied in the `/MigrationSnapshots/ProcServer` directory: Linux UNIX Windows

- Linux UNIX `BPMigrateCluster.sh /MigrationSnapshots/ProcServer applicationCluster1 dmgrProfile`
- Windows `BPMigrateCluster.bat c:\MigrationSnapshots\ProcServer applicationCluster1 dmgrProfile`

For more information about the `BPMigrateCluster` command, see the `BPMigrateCluster` command-line utility topic.

32. Enable synchronization for all clustered nodes.

Enable synchronization for all nodes in the cluster (both group A and group B) using the administrative console on the target deployment manager. To do this, use the following procedure.

- a. From the WebSphere Application Server administrative console, select **System administration > Node agents**.
 - b. Click the node agent for the node.
 - c. Click **File synchronization service**.
 - d. Select **Enable service at server startup, Automatic synchronization** and **Startup synchronization**.
 - e. Click **Apply**, then click **OK** to save the configuration changes.
33. Back up the group A migration source profiles.
 Repeat this step for each profile that will be migrated in group A. This backup is needed in case the next step for running the `syncNode` command fails. After resolving the issue with `syncNode`, you can restore the backup before running `syncNode` command again.
 Back up the profile configuration on the non-clustered managed node using the `backupConfig` command.
 Use the following syntax to back up a profile named `profile1` to `/ProfileBackups/profile1.zip`.

| | | |
|-------|------|---------|
| Linux | UNIX | Windows |
|-------|------|---------|

 - `backupConfig.sh /ProfileBackups/profile1.zip -profileName profile1`
 - `backupConfig.bat c:\ProfileBackups\profile1.zip -profileName profile1`
 For more information about the `backupConfig` command, see the `backupConfig` command topic on the WebSphere Application Server, Version 7.0 information center.
34. Synchronize all Group A nodes.
 Repeat this step for each group A clustered managed node in the cluster.
 Synchronize the node with the target deployment manager using the **`syncNode`** command from the `profile_root/bin` directory of the migration target profile or from the First steps console of the target profile.
 Use the following syntax:

| | | |
|-------|------|---------|
| Linux | UNIX | Windows |
|-------|------|---------|

 - `syncNode.sh deployment_manager_machine_name_or_ip_address deployment_manager_port_no`
 - `syncNode.bat deployment_manager_machine_name_or_ip_address deployment_manager_port_no`
 For more information about the **`syncNode`** command, see the `syncNode` command topic on the WebSphere Application Server, Version 7.0 information center.
35. Back up the cluster-scoped product databases configured for the cluster.
 Back up the following product databases that are configured for the cluster according to the documentation for your database:
36. Upgrade the product databases.
 Databases configured for the network deployment environment are automatically updated as part of starting the server in the following scenario:

Note: If you are migrating from version 7.0 or 7.5 to 7.5.1, it is not necessary to perform a schema upgrade for the Common database. Refer to “Migrating databases” on page 356.

Note: Refer to “Databases” on page 303 for the database permissions needed to perform schema upgrades.

The Common Event Infrastructure database and Messaging Engine database are automatically migrated by the runtime migration procedure when the profiles are migrated. For more information, see “Databases” on page 303.

In all other scenarios, the product databases must be upgraded manually. Refer to “Manually upgrading the product databases” on page 356.

If the source version is 7.0.x or 7.5, refer to Step 37 b to migrate the Business Space database schema. Business Space combines the schema upgrade and data migration into one procedure.

Optional: Migrate the messaging engine database if it is needed for your environment. To learn more about when and how to migrate the messaging engine, see the Migrating a messaging engine based on a data store topic on the WebSphere Application Server, Version 7.0 information center.

37. Migrate the instance data for Business Space if they are configured in the source environment. Perform the following steps to update the data in the databases to work with version 7.5.1.
 - a. If the source version is 6.x, migrate the Business Space database data using the procedure described in Migrating the Business Space database data (V6.x). If the source version is 7.0.x or 7.5, migrate the Business Space database schema using the procedure described in Migrating the Business Space V7.0.x or V7.5 database and data.

38. Start the group A migration target node agent.

Repeat this step for each group A clustered managed node in the cluster.

Start the migration target node agent using the `startNode` command from the `profile_root/bin` directory of the migration target server or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startNode.sh`
- Windows `startNode.bat`

For more information about the `startNode` command, see the `startNode` command topic in the WebSphere Application Server, Version 7.0 information center.

39. Start the group A migration target servers.

Repeat this step for each server associated with a group A clustered managed node in the cluster.

Start the migration target server using the `startServer` command from the `profile_root/bin` directory of the migration target server or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startServer.sh server_name`
- Windows `startServer.bat server_name`

For more information about the `startServer` command, see the `startServer` command topic in the WebSphere Application Server, Version 7.0 information center.

40. Back up the group B migration source profiles.

Repeat this step for each profile that will be migrated in group B.

Back up the profile configuration on the non-clustered managed node using the `backupConfig` command.

Use the following syntax to back up a profile named `profile1` to `/ProfileBackups/profile1.zip`. Linux UNIX Windows

- Linux UNIX `backupConfig.sh /ProfileBackups/profile1.zip -profileName profile1`
- Windows `backupConfig.bat c:\ProfileBackups\profile1.zip -profileName profile1`

For more information about the `backupConfig` command, see the `backupConfig` command topic in the WebSphere Application Server, Version 7.0 information center.

41. Migrate the group B managed nodes.

This step should be repeated for each group B managed node in the cluster.

- For a side-by-side migration, you can use the WebSphere ESB profile migration wizard or the WebSphere ESB migration command-line utilities to migrate the source profile.
 - To use the WebSphere ESB profile migration wizard, follow the Migrating a profile using the BPM profile migration wizard procedure on the system containing the source profile.
 - To use the WebSphere ESB migration command-line utilities, follow the Migrating a profile using the BPM migration command-line utilities procedure on the system containing the source profile.
- For a remote migration, follow the “Migrating a profile to a remote system” on page 351 procedure.
- For an operating system upgrade migration, follow the “Migrating a server while upgrading an operating system” on page 353 procedure.

Note: If you are migrating from IBM Business Process Manager V6.0.2, you must use the Migrating a profile using the BPM migration command-line utilities procedure.

42. Start the group B migration target node agent.

Repeat this step for each group B clustered managed node in the cluster.

Start the migration target node agent using the `startNode` command from the `profile_root/bin` directory of the migration target server or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startNode.sh`
- Windows `startNode.bat`

For more information about the `startNode` command, see the `startNode` command topic in the WebSphere Application Server, Version 7.0 information center.

43. Start the group B migration target servers.

Repeat this step for each server associated with a group B clustered managed node in the cluster.

Start the migration target server using the `startServer` command from the `profile_root/bin` directory of the migration target server or from the First steps console for the profile.

Use the following syntax: Linux UNIX Windows

- Linux UNIX `startServer.sh server_name`
- Windows `startServer.bat server_name`

For more information about the `startServer` command, see the `startServer` command topic in the WebSphere Application Server, Version 7.0 information center.

44. Optional: Uninstall the source deployment manager.

Once the migration is complete, the migration source deployment manager can be uninstalled.

45. Remove Compatibility Mode.

If you chose the compatibility option (which is the default), and if all your nodes are completely migrated to the target version, run the `convertScriptCompatibility` script from the `install_root/bin` directory on the deployment to remove compatibility.

Use the following syntax: `Linux` `UNIX` `Windows`

- `Linux` `UNIX` `convertScriptCompatibility.sh`
- `Windows` `convertScriptCompatibility.bat`

For more information about the `convertScriptCompatibility` command, see the `convertScriptCompatibility` command topic on the WebSphere Application Server, Version 7.0 information center.

Results

The network deployment environment is migrated to the target version.

What to do next

Verify that the migration was successful. For instructions, see “Verifying migration” on page 359.

Related concepts:

“Migration overview” on page 289

The process of moving applications, configuration, and databases from an earlier version of WebSphere ESB to this version of WebSphere ESB is referred to as version-to-version migration, or simply migration.

“Runtime premigration checklist” on page 309

Before you begin the process of migrating to a new version of WebSphere ESB, you should verify each of the items in this checklist.

Related tasks:

“Migrating a profile using the profile migration wizard”

The profile migration wizard is a graphical user interface (GUI) that guides you through the process of migrating a profile. Migrating a profile is just one step in a series of steps required to migrate a stand-alone or network deployment environment.

“Migrating a profile using the command-line utilities” on page 349

Use this subprocedure for migrating a profile using the command-line utilities.


“Manually upgrading the product databases” on page 356

Use this procedure to upgrade product databases manually.

“Verifying migration” on page 359

Verify that your migration was successful by checking the log files and the error files for each migration step, and checking operation with the administrative console.

Related reference:

 `installBRManager` command-line utility

The **`installBRManager`** command migrates the Business Rules Manager.

Runtime migration subprocedures

Use the runtime migration subprocedures as part of the process of performing a version-to-version migration.

Migrating a profile using the profile migration wizard

The profile migration wizard is a graphical user interface (GUI) that guides you through the process of migrating a profile. Migrating a profile is just one step in a series of steps required to migrate a stand-alone or network deployment environment.

About this task

This procedure describes the steps necessary to use the IBM Business Process Manager profile migration wizard to migrate a profile.

Procedure

1. Invoke the migration wizard.

Invoke the migration wizard using the `BPMigrate` command from the `target_install_root/bin/bpm_migration`

Use the following syntax:

- `Linux` `UNIX` `BPMigrate.sh`
- `Windows` `BPMigrate.bat`

For more information about the `BPMigrate` command, see the `BPMigrate` command-line utility topic.

2. Read the Welcome screen.

On the Business Process Management Profile Migration Wizard Welcome screen, read the information on the panel to learn about the migration process, and then click **Next**.

3. Select the wizard migration type: Typical or Custom.

On the Select Typical or Custom migration screen, select either a typical wizard migration or a custom wizard migration, and then click **Next**.

- If you select **Typical**, the migration wizard migrates the WebSphere ESB profile with the default configuration settings.
- If you select **Custom**, the migration wizard allows you to customize the configuration settings.

The default configuration settings are:

• **Snapshot directory:**

- **Linux** **UNIX** /MigrationSnapshots/*source_install_root*
- **Windows** C:\MigrationSnapshots*source_install_root*

- **Target profile name:** The default for the target profile name is the source profile name
- **Target profile directory:** The default for the target profile directory is the *target_install_directory/profiles/source_profile_name* where *source_profile_name* is the name of the source profile
- **Port value assignments:** Same as source profile port assignments
- **Script compatibility (deployment manage profiles only):** Set to true, so scripts from the source profile are still available post migration
- **Application directory settings (deployment manager profiles only):** Default target installation directory of the target profile

4. Select the source installation.

On the Select an installation to use as the source of the migration screen, select the source installation directory from the list of detected IBM Business Process Manager products or select **Browse** to select the installation directory of IBM Business Process Manager products not detected, and then click **Next**.

Restriction: If you are migrating from WebSphere ESB V6.0.2.x, you must use the “Migrating a profile using the command-line utilities” on page 349 procedure.

5. Select the source profile.

In the Select a source profile to use as the source of the migration screen, select the source profile from the list, enter the username and password if the profile has security enabled, then click **Next**.

6. Define the custom settings or skip to the verify step for a typical migration.

Note: If you selected **Typical** in Step 3, skip to Step 7 on page 348.

If you selected **Custom** in Step 3, use the following steps.

a. Select the snapshot directory.

On the Enter or browse for the snapshot directory to use for the source profile screen, keep the default snapshot directory or click **Browse** to navigate to a new snapshot directory, then click **Next**.

b. Specify the target profile name and target profile directory.

On the Select the target profile name and directory screen, keep the default target profile name and directory, or enter a new target profile name and directory in the **Target profile name** and **Target profile directory** fields, then click **Next**.

- c. Select the application migration setting.

Note: This screen appears only if you are migrating a deployment manager profile.

On the Select the application migration setting screen, specify where the migrated applications should be located and then click **Next**. The default selection is: **Install the applications in the default directory of the target installation**.

- **Install the applications in the default directory of the target installation.**
- **Keep the current application installation directories.**

Restrictions: If you choose this option, the location is shared by the existing installation and the new installation. If you keep the migrated applications in the same locations as those of the earlier version, the following restrictions apply:

- Mixed-node support limitations must be followed. This means that the following support cannot be used when invoking the **wsadmin** command:
 - Precompile JSP
 - Use Binary Configuration
 - Deploy EJB
- You risk losing the migrated applications unintentionally if you later delete applications from these locations when administering (uninstalling, for example) your previous installation.

- d. Select the port migration setting.

Note: This screen appears only if you are migrating a stand-alone profile.

On the Select the port migration setting screen, select one of the following options for assigning target profile port values, and then click **Next**.

- **Use the same port assignments as the source profile.**
- **Do not override the ports that were created with the target profile.**
- **Assign available ports to the target profile beginning with the following port number:**

If you select this option, enter the first value of the block of consecutive port numbers to assign.

Note: The default selection is: **Use the same port assignments as the source profile**.

- e. Select the script compatibility setting.

Note: This screen appears only if you are migrating a deployment manager profile.

On the Select the script compatibility setting screen, select or clear the **Enable source profile administrative scripts for use in the target installation** box, then click **Next**. Selecting this option sets the optional WebSphere Application Server `-scriptCompatibility` parameter to true.

Setting this parameter to true enables the migration to create the following WebSphere Application Server version 6.x configuration definitions:

- Transport
- ProcessDef
- Version 6.x SSL

instead of the following WebSphere Application Server 7.0 configuration definitions:

- Channels
- ProcessDefs
- Version 7.0 SSL

Select this option in order to minimize impacts to existing administration scripts. For example, if you have existing **wsadmin** scripts or programs that use third-party configuration APIs to create or modify the version 6.x configuration definitions, select this option.

Note: This is meant to provide a temporary transition until all of the nodes in the environment are at the WebSphere Application Server 7.0 level. When they are all at 7.0, you should perform the following actions:

- 1) Modify your administration scripts to use all of the 7.0 settings
- 2) Use the **convertScriptCompatibility** command to convert your configurations to match all of the 7.0. For more information, see the **convertScriptCompatibility** command topic in the WebSphere Application Server information center.

Note: When following the directions at this link to use the **convertScriptCompatibility** command, use the **BPMigrateProfile** command rather than the **WASPostUpgrade** command.

7. Verify the migration wizard selections.

On the Profile migration summary screen, verify the migration selections you made in the wizard, then click **Next** to begin the migration.

8. Monitor the status of the migration.

The Migration execution screen displays the status of the profile migration. Monitor the migration to validate that it is running successfully.

9. Retry the migration if it fails.

If the profile migration fails while copying the source profile, creating the target profile, or migrating the source profile to the target profile, use the following procedure to retry the migration.

- a. Fix the root cause of the failure.
- b. Remove the following artifacts created by the failed migration:
 - The snapshot directory
 - The target profile (using the **manageprofiles** command-line utility).

Note: If a deployment manager profile was being migrated, and the source deployment manager has been disabled, it should be re-enabled using the **migrationDisablementReversal** command to rollback the migration. However, if the profile migration is going to be re-executed, reversing the disablement of the deployment manager is unnecessary.

- c. Use the back button or restart the wizard to run the migration again.

10. Click **Next** if the migration completed successfully, and click **Finish** to exit the wizard.

Results

The profile is migrated from an earlier version of WebSphere ESB to WebSphere ESB V7.5.1.

What to do next

Verify that the migration was successful. For instructions, see “Verifying migration” on page 359.

Related tasks:

“Migrating a network deployment environment with full downtime” on page 322
Use this procedure to migrate a network deployment environment while incurring full downtime.

“Migrating a network deployment environment with minimal downtime” on page 330

Use this procedure to migrate a network deployment environment while incurring minimal downtime.

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

Migrating a profile using the command-line utilities

Use this subprocedure for migrating a profile using the command-line utilities.

Procedure

Follow these steps to migrate a profile using the command-line utilities.

1. Create a copy of the source profile.

Create a copy of the configuration files in the source profile that will be migrated to the target profile using the `BPMSnapshotSourceProfile` command from the `install_root/bin` directory. The user-specified snapshot directory should not be located in the source or target product installation directories, so those directories can be removed later if necessary without impacting the configuration files in the snapshot directory.

Use the following syntax to copy a source profile named `sourceProfile1` located in the `ProcServer620` installation root directory to the `/MigrationSnapshots/`

`ProcServer620` snapshot directory:

- Linux `BPMSnapshotSourceProfile.sh` /opt/ibm/WebSphere/ProcServer620 sourceProfile1 /MigrationSnapshots/ProcServer620
- Windows `BPMSnapshotSourceProfile.bat` "C:\Program Files\IBM\WebSphere\ProcServer620" sourceProfile1 c:\MigrationSnapshots\ProcServer620

For more information about the `BPMSnapshotSourceProfile` command, see the `BPMSnapshotSourceProfile` command topic.

2. Create the target profile.

Create the target profile using the `BPMCreateTargetProfile` command from the `install_root/bin` directory. This profile will not be ready for use until the `BPMmigrateProfile` command is used to migrate the source profile to the new target profile.

Use the following syntax to create a target profile for the migration using the source profile named `sourceProfile1` copied to the `/MigrationSnapshots/`

`ProcServer620` snapshot directory.

Linux Windows UNIX

- **Linux** **UNIX** `BPMCreateTargetProfile.sh /MigrationSnapshots/ProcServer620 sourceProfile1`
- **Windows** `BPMCreateTargetProfile.bat "C:\MigrationSnapshots\ProcServer620" sourceProfile1`

For more information about the `BPMCreateTargetProfile` command, see the `BPMCreateTargetProfile` command topic.

3. Migrate the source profile to the target profile.

Migrate the source profile to the target profile using the `BPMMigrateProfile` command. This command reads the configuration information from the snapshot directory specified by the `BPMSnapshotSourceProfile` command and migrates it to the target profile.

Use the following syntax to migrate the source profile named `sourceProfile1` copied into the `/MigrationSnapshots/ProcServer620` directory to the target profile:

- **Linux** **UNIX** `BPMMigrateProfile.sh /MigrationSnapshots/ProcServer620 sourceProfile1`
- **Windows** `BPMMigrateProfile.bat C:\MigrationSnapshots\ProcServer620 sourceProfile1`

If the source profile has security enabled, the `-username` and `-password` parameters are required and the user name provided must be a member of the operator or administrator role.

Windows On the Windows operating system, even if security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the system.

For more information about the `BPMMigrateProfile` command, see the `BPMMigrateProfile` command topic.

CAUTION:

On HP-UX and Solaris 64-bit operating systems, to avoid `OutOfMemory` errors during profile migration, determine whether the default JVM heap size of 768 MB will work. Depending on the number of applications that are part of the profile being migrated, this parameter might need adjustment. Use the `-javaoption` of the `BPMMigrateProfile` command-line utility to override the default value and specify a custom value for JVM heap size.

4. Check the migration status.

Use the `BPMMigrationStatus` command to verify the current state of the migration.

Use the following syntax:

- **Linux** **UNIX** `BPMMigrationStatus.sh`
- **Windows** `BPMMigrationStatus.bat`

For more information about the `BPMMigrationStatus` command, see the `BPMMigrationStatus` command topic.

Results

The profile is migrated from an earlier version of WebSphere ESB to WebSphere ESB V7.5.1.

What to do next

Verify that the migration was successful. For instructions, see “Verifying migration” on page 359.

Related tasks:

“Migrating a network deployment environment with full downtime” on page 322
Use this procedure to migrate a network deployment environment while incurring full downtime.

“Migrating a network deployment environment with minimal downtime” on page 330

Use this procedure to migrate a network deployment environment while incurring minimal downtime.

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

Migrating a profile to a remote system

Use this procedure for migrating a server profile to a remote system.

Before you begin

Important: You must install WebSphere ESB version 7.5.1 on a machine with the same operating system as that of the machine where the source installation is located. The **BPMCreateRemoteMigrationUtilities** command creates a .zip file that works only when the command is executed on the same operating system. When the snapshot step has completed, you can use the created .zip file on a different machine with a different operating system.

Note: If the profile that is being migrated is a federated profile, synchronize the profile with the new version 7.5.1 deployment manager before attempting migration. See Manually synchronize all nodes in the Websphere Application Server information center.

Procedure

Follow the steps in this procedure to migrate a profile to a remote system.

1. Create a default profile on the target system. On the migration target system, create a default profile; for example, by using the instructions in Creating stand-alone profiles using the Profile Management Tool.
2. Create the remote migration utilities image.

On the migration target system, or any system that has version 7.5.1 installed, create a remote migration image using the **BPMCreateRemoteMigrationUtilities** command from the *install_root/bin* directory.

Use the following syntax: Linux Windows UNIX

- Linux UNIX **BPMCreateRemoteMigrationUtilities.sh**
remoteMigrationUtilities.zip
- Windows **BPMCreateRemoteMigrationUtilities.bat**
remoteMigrationUtilities.zip

For more information about the **BPMCreateRemoteMigrationUtilities** command, see the **BPMCreateRemoteMigrationUtilities** command topic.

3. Copy the remote migration utilities to the source system.

Using ftp, rcp, or some other mechanism, copy the remote migration utilities from the target system to the source system and unzip the remote migration utilities on the source system into their own unique directory.

Linux **UNIX** On UNIX platforms, use the **tar -xvf** command to extract the contents of the .zip file.

4. Create a snapshot of the migration source profile.

On the migration source system, use the **BPMSnapshotSourceProfile** command from the remote migration utilities bin directory to create a snapshot directory containing the configuration files that will be migrated. Use the option **-remoteMigration** to indicate that the target profile will not be created on the same machine. This option saves additional configuration information from the source profile that will be used during remote migration.

Use the following syntax with the **-remoteMigration** option to snapshot a source profile named **sourceProfile1** located in the **ProcServer700** installation root directory to the **/MigrationSnapshots/ProcServer700** snapshot directory:

Important: **Linux** **UNIX** On a UNIX system, ensure that all extracted files have execute permission for the logged-in user. If not, use the **chmod** command to grant execute permission for all extracted files.

For more information about the **BPMSnapshotSourceProfile** command, see the **BPMSnapshotSourceProfile** command topic.

5. Copy the migration source snapshot directory to the migration target system. Create a .zip of the source snapshot directory, copy it to the same directory on the target system, and unzip it there.
6. Create the target profile.

Create the target profile using the **BPMCreateTargetProfile** command. This profile will not be ready for use until the **BPMigrateProfile** command is used to migrate the source profile to the new target profile.

Use the following syntax to create a target profile for the migration using the source profile named **sourceProfile1** copied to the **/MigrationSnapshots/ProcServer700** snapshot directory. **Linux** **Windows** **UNIX**

- **Linux** **UNIX** **BPMCreateTargetProfile.sh -remoteMigration true /MigrationSnapshots/ProcServer700 sourceProfile1**
- **Windows** **BPMCreateTargetProfile.bat -remoteMigration true C:\MigrationSnapshots\ProcServer700 sourceProfile1**

For more information about the **BPMCreateTargetProfile** command, see the **BPMCreateTargetProfile** command topic.

7. Migrate the source profile to the target profile.

Migrate the source profile to the target profile using the **BPMigrateProfile** command. This command will read the configuration information from the snapshot directory specified by the **BPMSnapshotSourceProfile** command and copied over to the target system and migrate it to the target profile.

Use the following syntax to migrate the source profile named **sourceProfile1** copied into the **/MigrationSnapshots/ProcServer700** directory to the target profile: **Linux** **Windows** **UNIX**

- **Linux** **UNIX** **BPMigrateProfile.sh /MigrationSnapshots/ProcServer700 sourceProfile1**
- **Windows** **BPMigrateProfile.bat C:\MigrationSnapshots\ProcServer700 sourceProfile1**

If the source profile does not have security enabled the username and password parameters are unnecessary, otherwise, the user name provided must be a member of the operator or administrator role.

Windows On the Windows operating system, even if security is enabled, the `-username` and `-password` parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the system. For more information about the **BPMigrateProfile** command, see the **BPMigrateProfile** command topic.

8. Check the migration status.

Use the **BPMigrationStatus** command to verify the current state of the migration.

- **Linux** **UNIX** **BPMigrationStatus.sh**
- **Windows** **BPMigrationStatus.bat**

For more information about the **BPMigrationStatus** command, see the **BPMigrationStatus** command topic.

9. Scan the file system under the profile directory for occurrences of old hostname value. Analyze the configuration where the old hostname is still being used and replace it with new hostname, unless the old hostname is needed, such as of the database is still present on the old hostname machine.

Results

The profile is migrated from an earlier version of WebSphere ESB to WebSphere ESB V7.5.1 on a remote system.

What to do next

Verify that the migration was successful. For instructions, see “Verifying migration” on page 359.

Related tasks:

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

Migrating a server while upgrading an operating system

User the following procedure for migrating a server profile on a system whose operating system is being upgraded.

Before you begin

See the topic.

Procedure

Follow the steps in this procedure to migrate a profile on a system whose operating system is being upgraded.

1. **Create the remote migration utilities image.**

From any system that has a version 7.5.1 installed, create a remote migration utilities image using the **BPMCreateRemoteMigrationUtilities** command from the *install_root/bin* directory.

Use the following syntax:

- **Linux** **UNIX** `BPMCreateRemoteMigrationUtilities.sh`
`remoteMigrationUtilities.zip`
- **Windows** `BPMCreateRemoteMigrationUtilities.bat`
`remoteMigrationUtilities.zip`

For more information about the **BPMCreateRemoteMigrationUtilities** command, see the **BPMCreateRemoteMigrationUtilities** command topic.

2. Copy the remote migration utilities to the source system.

Using ftp, rcp, or some other mechanism, copy the remote migration utilities from the target system to the source system and unzip the remote migration utilities on the source system into their own unique directory.

3. Snapshot the migration source profile.

On the migration source system use the **BPMSnapshotSourceProfile** command from the remote migration utilities bin directory to create a snapshot directory containing the configuration files that will be migrated.

Use the following syntax to snapshot a source profile named `sourceProfile1` located in the `ProcServer620` installation root directory to the

`/MigrationSnapshots/ProcServer700` snapshot directory: **Linux** **UNIX**

Windows

- **Linux** **UNIX** `BPMSnapshotSourceProfile.sh /opt/ibm/WebSphere/ProcServer700 sourceProfile1 /MigrationSnapshots/ProcServer700`
- **Windows** `BPMSnapshotSourceProfile.bat "C:\Program Files\IBM\WebSphere\ProcServer700" sourceProfile1 c:\MigrationSnapshots\ProcServer700`

For more information about the **BPMSnapshotSourceProfile** command, see the **BPMSnapshotSourceProfile** command topic.

4. Copy the migration source snapshot directory to a temporary location.

Create a zip of the source snapshot directory, copy it to a remote system temporarily while the source system is being upgraded.

5. Upgrade the source system's operating system.

Upgrade the system's operating system to the appropriate version.

6. Install the migration target product(s).

Install the target product and latest fix packs on the same system as the source product of the migration.

Note: To migrate from source profiles augmented by multiple products, the new version of those products must be installed into the same target installation directory. For example, if the source profile is augmented by IBM Business Process Manager and IBM Business Monitor, both of those products must be installed into the same target installation directory.

7. Restore the migration source snapshot directory.

Copy the snapshot directory .zip file that was stored temporarily on the remote system back to the freshly upgraded target migration system. Unzip the .zip file to create a snapshot directory folder on the target migration system. Use the unzipped root directory as the snapshot directory value for the remaining IBM Business Process Manager migration command-line utilities.

8. Create the target profile.

Create the target profile using the **BPMCreateTargetProfile** command. This profile will not be ready for use until the **BPMMigrateProfile** command is used to migrate the source profile to the new target profile.

Use the following syntax to create a target profile for the migration using the source profile named sourceProfile1 copied to the /MigrationSnapshots/

ProcServer700 snapshot directory. Linux UNIX Windows

- Linux UNIX `BPMCreateTargetProfile.sh /MigrationSnapshots/ProcServer700 sourceProfile1`
- Windows `BPMCreateTargetProfile.bat "C:\MigrationSnapshots\ProcServer700" sourceProfile1`

For more information about the **BPMCreateTargetProfile** command, see the **BPMCreateTargetProfile** command topic.

9. Migrate the source profile to the target profile.

Migrate the source profile to the target profile using the **BPMMigrateProfile** command. This command will read the configuration information from the snapshot directory specified by the **BPMSnapshotSourceProfile** command and migrate it to the target profile.

Use the following syntax to migrate the source profile named sourceProfile1 copied into the /MigrationSnapshots/ProcServer700 directory to the target

profile: Linux UNIX Windows

- Linux UNIX `BPMMigrateProfile.sh /MigrationSnapshots/ProcServer700 sourceProfile1`
- Windows `BPMMigrateProfile.bat "C:\MigrationSnapshots\ProcServer700" sourceProfile1`

If the source profile does not have security enabled the -username and -password parameters are unnecessary; otherwise, the user name provided must be a member of the operator or administrator role.

On the Windows operating system, even if security is enabled, the -username and -password parameters do not have to be specified if the server is running as a Windows service. In this case, the parameters are automatically passed into the script that the Windows service uses to shut down the system.

For more information about the **BPMMigrateProfile** command, see the **BPMMigrateProfile** topic.

10. Check the migration status.

Use the **BPMMigrationStatus** command to verify the current state of the migration. Linux UNIX Windows

- Linux UNIX `BPMMigrationStatus.sh`
- Windows `BPMMigrationStatus.bat`

For more information about the **BPMMigrationStatus** command, see the **BPMMigrationStatus** command topic.

Results

The profile is migrated from an earlier version of WebSphere ESB to WebSphere ESB V7.5.1, and the operating system is upgraded.

What to do next

Verify that the migration was successful. For instructions, see “Verifying migration” on page 359.

Related tasks:

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

Migrating databases

Databases configured for a stand-alone environment are automatically updated as part of starting the server in some scenarios. In other scenarios, they must be upgraded manually.

Databases configured for a stand-alone environment are automatically updated as part of starting the server in the following scenario:

- Common database: The database user configured for the common database has the needed permissions.

Note: Refer to “Databases” on page 303 for the database permissions needed to perform schema upgrades.

The Common Event Infrastructure database and Messaging Engine database are automatically migrated by the runtime migration procedure when the profiles are migrated. For more information, see “Databases” on page 303.

To upgrade product databases that are not updated automatically, follow the “Manually upgrading the product databases” procedure.

Manually upgrading the product databases

Use this procedure to upgrade product databases manually.

About this task

Use this procedure to upgrade the product databases if they are not updated automatically as part of starting the server.

Procedure

1. Run the `BPMCreateDatabaseUpgradeUtilities` command-line utility to create a .zip file to copy files needed for generation of upgrade SQL scripts. The .zip file contains files from WebSphere ESB installation, along with some configuration from the *snapshot_directory*.
2. Extract the contents of the .zip file to the database computer where the generated SQL scripts for a given database can be run. Alternatively, you can copy the generated scripts from step 3 to a database computer, from which upgrades can be performed using an SQL session, once the scripts have been generated.
3. Run the `BPMGenerateUpgradeSchemaScripts` command-line utility to generate SQL scripts for each database that is to be upgraded.

To obtain a list of databases, open the `DatabaseInfo.txt` file in the following location:

- `Linux` `UNIX` *unzipped location/snapshot_directory*
- `Windows` *unzipped location\snapshot_directory*

The DatabaseInfo.txt file lists the deployment targets with the respective component mapping to a database schema. The values are defined as *name-value* pairs, where *name* is *DeploymentTargetType_DeploymentTargetName_ComponentName* and *value* is *DatabaseName.SchemaName*.

To upgrade the database for a specific component, check for the component name in the first column and run the script against the respective *DatabaseName.SchemaName*.

The SQL scripts are generated in the following locations:

- **Linux** **UNIX** *Unzipped Location/snapshot_directory/DB Type/Database name.Schema name*
 - **Windows** *Unzipped Location\snapshot_directory\DB Type\Database name.Schema name*
4. Run the SQL scripts. Use one of these methods:
 - Run the SQL scripts using the **upgradeSchema.bat** or **upgradeSchema.sh** file that was generated along with the SQL scripts. Refer to BPMGenerateUpgradeSchemaScripts command-line utility for information about the command.
 - Run the SQL scripts using an SQL session with special configuration. Refer to “Running SQL upgrade scripts” on page 305.
 5. Check the result.log file that is generated during execution for errors.

What to do next

If you have a Business Space database, after you complete this task, you must migrate the database data.

Important: If you are migrating Business Space from V7.0.x or V7.5, a separate procedure is required to migrate both the Business Space database schema and data to V7.5.1.

Related concepts:

“Running SQL upgrade scripts” on page 305

A database administrator (DBA) can run an SQL upgrade script by invoking the upgradeSchema.bat or the upgradeSchema.sh script or by running the SQL script directly.

Related tasks:

“Migrating a network deployment environment with full downtime” on page 322
Use this procedure to migrate a network deployment environment while incurring full downtime.

“Migrating a network deployment environment with minimal downtime” on page 330

Use this procedure to migrate a network deployment environment while incurring minimal downtime.

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

Related information:

 BPMGenerateUpgradeSchemaScripts command-line utility

Migrating security

You can migrate the security configuration and security settings from a previous release of the product to WebSphere ESB V7.5.1.

Migrating the security configuration for a stand-alone environment

You can migrate the security configuration and security settings from a previous version of WebSphere ESB to the current version.

Before you begin

If security is enabled in the previous release, obtain the administrative user ID and password for the server where the previous release is installed. This information is needed in order to perform migration tasks. The security settings from the previous release will be overwritten during migration with the default WebSphere ESB security configuration. Make a note of any non-default security settings that are enabled in the previous release so that you can reconfigure the settings after migration.

About this task

Follow these steps to migrate the security configuration from a previous version of WebSphere ESB to the current version:

Procedure

1. Migrate the stand-alone profile as described in Migrating a profile using the profile migration wizard
2. Augment the stand-alone profile as described in Augmenting profiles.
3. Review the postmigration tasks as described in Postmigration tasks.
4. Reconfigure the security settings after migration. The migrated security configuration is overwritten by the security configuration from the current version of WebSphere ESB.
5. Add the new default users for WebSphere ESB to the custom-based federated user repository. The default users bpmAuthor and admin are required for administration of WebSphere ESB.

Related tasks:

“Postmigration tasks for WebSphere ESB” on page 366

After migration, you might need to check some configuration settings, or further configure the V7.5.1 server.

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

“Augmenting profiles” on page 174

You can augment an existing profile for WebSphere Application Server version 7.0 or WebSphere Application Server Network Deployment version 7.0 to add support for WebSphere Enterprise Service Bus.

Migrating the security configuration for a network deployment environment

You can migrate the security configuration and security settings from a previous version of WebSphere ESB to the current version.

Before you begin

If security is enabled in the previous release, obtain the administrative user ID and password for the server where the previous release is installed. This information is needed in order to perform migration tasks. The security settings from the previous release will be overwritten during migration with the default WebSphere ESB security configuration. Make a note of any non-default security settings that

are enabled in the previous release so that you can reconfigure the settings after migration.

About this task

Follow these steps to migrate the security configuration from IBM Process Server to WebSphere ESB Advanced Edition:

Procedure

1. Migrate the deployment manager profile as described in the topic Migrating a profile using the profile migration wizard.
2. Migrate the profile for each managed node as described in the topic Migrating a profile using the profile migration wizard.
3. Upgrade the product databases as described in the topic Manually upgrading the product databases.
4. Review the postmigration tasks as describe in the related topic Postmigration tasks.

Related tasks:

“Postmigration tasks for WebSphere ESB” on page 366

After migration, you might need to check some configuration settings, or further configure the V7.5.1 server.

Verifying migration

Verify that your migration was successful by checking the log files and the error files for each migration step, and checking operation with the administrative console.

Before you begin

Make sure the server that has been migrated has been started.

Procedure

1. Execute the **BPMMigrationStatus** command for each machine that participated in the migration process. This command displays the status of each migration step and whether it was successful or failed.
2. Check the migration log files for any failures or messages.
 - *backupDirectory*/logs/
BPMSnapshotSourceProfile.*profileName.timestamp*.log
 - *backupDirectory*/logs/WASPreUpgrade.*timestamp*.log
 - *backupDirectory*/logs/BPMCreateTargetProfile.ProcSrvMig01.*timestamp*.log
 - *backupDirectory*/logs/BPMMigrateProfile.*profileName.timestamp*.log
 - *backupDirectory*/logs/BPMProfileUpgrade.*profileName.timestamp*.log
 - *backupDirectory*/logs/WASPostUpgrade.*profileName.timestamp*.log
 - *backupDirectory*/logs/BPMMigrateCluster.ant.*profile_name.timestamp*.log

Note: *backupDirectory* is the directory in which migrated data was first stored and later retrieved from during the migration process, as specified in the migration wizard or the **BPMSnapshotSourceProfile** or **BPMMigrateProfile** commands.

Note: *profileName* is the name of the new profile you created in V7.5.1 of WebSphere ESB.

All of these log files must indicate success, as described by these messages, for you to consider the migration successful.

3. Check the profile's log files for fatal profile creation or augmentation errors. Profile log files are located in the following directory: *install_root/logs/manageprofiles*. The log files contain the profile name in them, for example: create <profile name>.log.
4. Check the server log files.
 - a. Navigate to the *profile_root/logs/server_name* directory corresponding to the migrated profile.
 - b. Review the SystemOut.log file and make sure there are no fatal errors.
 - c. Review the SystemErr.log file and make sure there are no fatal errors.
5. Verify the Common database upgrade. If the Common database upgrade was not performed manually because the user configured for WebSphere ESB has all the necessary permissions, check that the database was upgraded successfully during deployment manager startup.
 - a. Navigate to the profile directory for the deployment manager. Typically this is *install_root/profiles/<profile name>*.
 - b. Navigate to the *logs* folder and check the SystemOut.log file. Look for the messages The Common Database Schema upgrade is started and CWLDB0003I: IBM Process Server Schema version was updated to "7.5.1" successfully.
6. Check the error files.
 - a. Navigate to the snapshot directory and check the error files. Errors for each migration step are logged in separate files whose names begin with *commandname* and end with .error.
 - b. The error files should be empty. If you find an error, look in the log file and follow the sequence that leads to the error.
7. Check operation with the administrative console.
 - a. Open the administrative console (Integrated Solutions Console).
 - b. Select **Applications > Enterprise Applications** from the navigation panel.
 - c. In the right corner panel, verify that all of the applications listed have started, shown by the green "started" icon.
 - d. From the navigation panel, select .
 - e. For each WebSphere ESB data source listed on this panel, select the check box and then select **Test connection**.

Note: **Test connection** does not work for ME datasources. To verify the connection for ME datasources, make sure there are no errors in the logs after the servers are started.

- f. For each data source, you should receive a message similar to the following: "The test connection operation for data source on server Dmgr1 at node Dmgr1Node1 was successful".

What to do next

If migration was successful, you can begin using the server. If the migration was not successful, refer to "Runtime migration troubleshooting" on page 372 for troubleshooting information.

Related tasks:

“Migrating a network deployment environment with full downtime” on page 322
Use this procedure to migrate a network deployment environment while incurring full downtime.

“Migrating a network deployment environment with minimal downtime” on page 330

Use this procedure to migrate a network deployment environment while incurring minimal downtime.

“Migrating a stand-alone environment” on page 317

Use this procedure to migrate a stand-alone environment.

Rolling back your environment

After migrating to WebSphere ESB V7.5.1, you can roll back to the version you migrated from, which can be a V7.5, 7.0, 6.2.0, or 6.1.0 environment. This returns the configuration to the state that it was in before migration. After rolling back the environment, you can restart the migration process.

About this task

Generally, migration does not modify anything in the configuration of the prior release; however, there are cases where minimal changes are made that are reversible—those of a deployment manager and its managed nodes.

The following subtopics provide further information for these cases.

Rolling back a deployment cell

You can use the **restoreConfig** and **wsadmin** commands to roll back a migrated WebSphere ESB V7.5.1 deployment cell to V7.5, 7.0, 6.2.0, or 6.1.0. This returns the configuration to the state that it was in before migration. After rolling back the deployment cell, you can restart the migration process.

Before you begin

When migrating a V7.5, 7.0, 6.2.0, or 6.1.0 deployment cell, you must perform all of the following backup steps in a sequence to successfully complete the rollback.

1. Back up the databases that support WebSphere ESB components.

Important: If the purpose of the rollback is to fix a problem that occurred during migration and rerun profile migration, do not perform a database rollback. Databases should be rolled back only if you need to start the managed node servers once they have been restored to the previous version.

Roll back the deployment target-scoped database, if applicable. Check whether the managed node has a server with the Business Space component configured.

- If the managed node does not have the Business Space component configured on any server, go to Step 2.
- If the managed node contains a server-scoped configuration for Business Space, roll back the database.

Important: If the managed node has a server that is a member of a cluster where Business Space is configured, verify that rolling back the managed node would not result in a mixed-version cluster, with some cluster members on version 7.5.1 and some on a previous version. A database rollback would cause cluster members on version 7.5.1 to fail. Databases should be rolled

back only if you decide to roll back all the managed nodes that participate in the cluster, and you plan to run the cluster on the previous version.

2. Optional: Back up your existing configuration using the **backupConfig** command or your own preferred backup utility.
 - Run the **backupConfig** command or your own preferred utility to back up the V7.5, 7.0, 6.2.0, or 6.1.0 deployment manager configuration.

Important: Make sure that you note the exact name and location of this backed-up configuration.

See the backupConfig command on the WebSphere Application Server information center.

- Run the **backupConfig** command or your own preferred utility to back up the V7.5, 7.0, 6.2.0, or 6.1.0 managed node configurations.

Important: Make sure that you note the exact name and location of each of these backed-up configurations.

See the backupConfig command on the WebSphere Application Server information center.

3. Migrate the deployment cell.

Procedure

1. Stop all of the servers that are currently running in the WebSphere ESB V7.5.1 environment.
2. If you chose to disable the previous deployment manager when you migrated to the V7.5.1 deployment manager, do one of the following:
 - a. If you backed up your previous deployment manager configuration using the **backupConfig** command or your own preferred backup utility, run the **restoreConfig** command or your own preferred utility to restore the V7.5, 7.0, 6.2.0, or 6.1.0 configuration for the deployment manager.

Important: Make sure that you restore the same backed-up configuration that you created just before you migrated the deployment manager.

See the restoreConfig command on the WebSphere Application Server information center.

- b. If you did not back up your previous deployment manager configuration, use the **wsadmin** command to run the migrationDisablementReversal.jacl script from the V7.5, 7.0, 6.2.0, or 6.1.0 *profile_root/bin* directory of the deployment manager that you need to roll back from V7.5.1.

Linux In a Linux environment, for example, use the following parameters:

```
./wsadmin.sh -f migrationDisablementReversal.jacl -conntype NONE
```

Tip: If you have trouble running the migrationDisablementReversal.jacl script, try to go through the steps in the script manually.

- 1) Go to the following directory:

profile_root/config/cells/cell_name/nodes/node_name

where *node_name* is the name of the deployment manager node that you want to roll back.

- 2) If you see a serverindex.xml_disabled file in this directory, do the following:

- a) Delete or rename the `serverindex.xml` file.
 - b) Rename the `serverindex.xml_disabled` file to `serverindex.xml`.
- 3. For each of the deployment cell's managed nodes that you need to roll back, do one of the following:
 - a. If you backed up your previous managed node configuration using the **backupConfig** command or your own preferred backup utility, run the **restoreConfig** command or your own preferred utility to restore the V7.5, 7.0, 6.2.0, or 6.1.0 configuration for the managed node.

Important: Make sure that you restore the same backed-up configuration that you created just before you migrated the managed node.

See the `restoreConfig` command on the WebSphere Application Server information center.

- b. If you did not back up your previous managed node configuration, use the **wsadmin** command to run the `migrationDisablementReversal.jacl` script from the V7.5, 7.0, 6.2.0, or 6.1.0 *profile_root/bin* directory of the managed node.

Linux In a Linux environment, for example, use the following parameters:

```
./wsadmin.sh -f migrationDisablementReversal.jacl -conntype NONE
```

Tip: If you have trouble running the `migrationDisablementReversal.jacl` script, try to go through the steps in the script manually.

- 1) Go to the following directory:

```
profile_root/config/cells/cell_name/nodes/node_name
```

where *node_name* is the name of the managed node that you want to roll back.

- 2) If you see a `serverindex.xml_disabled` file in this directory, do the following:
 - a) Delete or rename the `serverindex.xml` file.
 - b) Rename the `serverindex.xml_disabled` file to `serverindex.xml`.
- 4. Synchronize the managed nodes if they were ever running when the V7.5.1 deployment manager was running.
See `syncNode` command on the WebSphere Application Server information center.
- 5. If you chose to keep the installed applications in the same location as the prior release during migration to V7.5.1 and any of the V7.5.1 applications are not compatible with the prior release, install applications that are compatible.
- 6. Delete the V7.5.1 profiles.
See `Deleting a profile` on the WebSphere Application Server information center.
- 7. Roll back your databases. (For any databases that support WebSphere ESB components that were upgraded, either automatically with the migration tools or manually, restore the backups that you made before you started the migration process.)
- 8. Start the rolled-back deployment manager and its managed nodes in the V7.5, 7.0, 6.2.0, or 6.1.0 environment.
- 9. Enable synchronization for all the nodes if it was disabled when you were following the steps in “Migrating a network deployment environment with minimal downtime” on page 330. To do this, use the following procedure.

- a. From the WebSphere Application Server administrative console, select **System administration > Node agents**.
- b. Click the node agent for the node.
- c. Click **File synchronization service**.
- d. Select **Enable service at server startup, Automatic synchronization and Startup synchronization**.
- e. Click **Apply**, then click **OK** to save the configuration changes.

Results

The configuration should now be returned to the state that it was in before migration.

What to do next

You can now restart the migration process if you want to do so.

Rolling back a managed node

You can use the **restoreConfig** and **wsadmin** commands to roll back a migrated WebSphere ESB V7.5.1 managed node to the state that it was in before migration. For each managed node that you want to roll back, you must roll back the managed node itself and the corresponding changes made to the master repository located on the deployment manager.

Before you begin

When you migrate a V7.5, 7.0, 6.2.0, or 6.1.0 managed node, you must perform all of the backup steps below in a sequence to successfully complete the rollback.

1. Back up the databases that support WebSphere ESB components.

Important: If the purpose of the rollback is to fix a problem that occurred during migration and rerun profile migration, do not perform a database rollback. Databases should be rolled back only if you need to start the managed node servers once they have been restored to the previous version.

Roll back the deployment target-scoped database, if applicable. Check whether the managed node has a server with the Business Space component configured.

- If the managed node does not have the Business Space component configured on any server, go to Step 2.
- If the managed node contains a server-scoped configuration for Business Space, roll back the database.

Important: If the managed node has a server that is a member of a cluster where Business Space is configured, verify that rolling back the managed node would not result in a mixed-version cluster, with some cluster members on version 7.5.1 and some on a previous version. A database rollback would cause cluster members on version 7.5.1 to fail. Databases should be rolled back only if you decide to roll back all the managed nodes that participate in the cluster, and you plan to run the cluster on the previous version.

2. Back up your existing configuration using the **backupConfig** command or your own preferred backup utility.
 - Run the **backupConfig** command or your own preferred utility to back up the V7.5, 7.0, 6.2.0, or 6.1.0 deployment manager configuration.

Important: Make sure that you note the exact name and location of this backed-up configuration.

See the **backupConfig** command in the WebSphere Application Server Network Deployment, version 6.1 information center.

- Run the **backupConfig** command or your own preferred utility to back up the V7.5, 7.0, 6.2.0, or 6.1.0 managed node configuration.

Important: Make sure that you note the exact name and location of this backed-up configuration.

See the **backupConfig** command in the WebSphere Application Server Network Deployment, version 6.1 information center.

3. Migrate the managed node.

If necessary, you can now roll back the managed node that you just migrated.

Important: If you do not have a backup copy of your V7.5.1 deployment manager configuration as it was before you migrated the V7.5, 7.0, 6.2.0, or 6.1.0 managed node that you want to roll back, you cannot use the procedure described in this article and you must roll back your whole cell as described in “Rolling back a deployment cell” on page 361.

About this task

You must perform all of the backup and rollback actions for each migrated managed node before you proceed to roll back another managed node.

Procedure

1. Roll back your databases. (For any databases that support WebSphere ESB components that were upgraded, either automatically with the migration tools or manually, restore the backups that you made before you started the migration process.)
2. Stop all of the servers that are currently running in the V7.5.1 environment.
3. Restore your previous configuration.
 - a. Run the **restoreConfig** command or your own preferred utility to restore the V7.5.1 deployment manager configuration.

Important: Make sure that you restore the same backed-up configuration that you created just before you migrated the managed node.

See the **restoreConfig** command in the WebSphere Application Server Network Deployment, version 6.1 Information Center.

- b. Perform one of the following actions to restore the V7.5, 7.0, 6.2.0, or 6.1.0 configuration for the managed node.
 - Run the **restoreConfig** command or your own preferred utility to restore the V7.5, 7.0, 6.2.0, or 6.1.0 configuration.
See **restoreConfig** command in the WebSphere Application Server Network Deployment, version 6.1 Information Center.
 - Use the **wsadmin** command to run the `migrationDisablementReversal.jacl` script from the V7.5, 7.0, 6.2.0, or 6.1.0 `profile_root/bin` directory of the managed node.

Linux In a Linux environment, for example, use the following parameters:

```
./wsadmin.sh -f migrationDisablementReversal.jacl -conntype NONE
```

Tip: If you have trouble running the `migrationDisablementReversal.jacl` script, try to manually perform the steps in the script.

1) Go to the following directory:

`profile_root/config/cells/cell_name/nodes/node_name`

where `node_name` is the name of the managed node that you want to roll back.

2) If you see a `serverindex.xml_disabled` file in this directory, perform the following actions:

a) Delete or rename the `serverindex.xml` file.

b) Rename the `serverindex.xml_disabled` file to `serverindex.xml`.

4. Start the V7.5.1 deployment manager.

5. Synchronize the managed node.

See Synchronizing nodes with the `wsadmin` tool in the WebSphere Application Server Network Deployment, version 6.1 information center.

6. If you chose to keep the installed applications in the same location as the prior release during migration to V7.5.1 and any of the V7.5.1 applications are not compatible with the prior release, install applications that are compatible.

7. Delete the V7.5.1 managed profile.

See Deleting a profile in the WebSphere Application Server Network Deployment, version 6.1 Information Center.

8. Start the rolled-back managed node in the V7.5.1 environment.

Results

The configuration should now be returned to the state that it was in before migration.

What to do next

You can now restart the migration process if you want to do so.

Postmigration tasks

Postmigration tasks are tasks you perform on WebSphere ESB and Business Space after successfully migrating to V7.5.1.

Postmigration tasks for WebSphere ESB

After migration, you might need to check some configuration settings, or further configure the V7.5.1 server.

Before you begin

Ensure that you have migrated your server or cluster and verified that the migration was successful.

About this task

Perform the following checks, if applicable to your environment:

- Examine any Lightweight Third Party Authentication (LTPA) security settings that you might have used in V7.5, 7.0, 6.2.0, or 6.1.0, and make sure that V7.5.1 security is set appropriately.

- Check the `BPMigrateProfile.profile_name.timestamp.log` file in the logs directory for details about any JSP objects that the migration tools did not migrate.

If V7.5.1 does not support a level for which JSP objects are configured, the migration tools recognize the objects in the output and log them.

- Review your Java virtual machine settings to verify that you are using the recommended heap sizes. See Java virtual machine settings. The information at this link applies to WebSphere ESB servers as well as to WebSphere Application Server servers.
- After migrating from V7.0.0.x to V7.5.1, check your WebSphere Adapter properties to ensure that they are properly configured for the new installation location. Some adapter properties might need to be altered during migration in a way that would be unknown to an automated migration.
- If you had uninstalled applications using Websphere Adapters before migration as mentioned in Runtime premigration checklist, you can use IBM Integration Designer to update the applications to use Websphere Adapter version 7.5.1 and install them in the 7.5.1 target environment. You can install these updated applications using the administrative console. See Migrating applications using previous adapter levels for more information.
- To support the migration of version 6.1.0.x, 6.1.2.x, or 6.2.0.x of the WebSphere Adapter for SAP to WebSphere ESB V7.5.1, you need to migrate the previous version of the application in IBM Integration Designer before it can be deployed onto WebSphere ESB V7.5.1.
- After migrating to V7.5.1, be aware that the default value for the target significance property has changed from V7.5, 7.0, 6.2.0, or 6.1.0. In V7.5.1, the default value was changed from `targetSignificance=preferred` to `targetSignificance=required`. The new default value is set in the JMS activation specifications and connection factories that are part of the WebSphere ESB configuration.

You must determine whether to change the target significance value in the migrated environment (V7.5, 7.0, 6.2.0, or 6.1.0).

- After migrating to V7.5.1, check that your ports are mapped correctly to make sure that the Remote Artifact Loader can access the security port on the application cluster when the global security is turned on. To verify that your ports are configured correctly, use the following procedure:
 1. In the administrative console, navigate to **Environment > Virtual Hosts**.
 2. Select **default_host > Host Aliases**.
 3. Check if the application cluster security port is mapped to "*" which means "all hosts." If it is not, change it to "*" by clicking **New**, then entering "*" in the **Host Name** field and the port number of the application cluster in the **Port** field.
 4. Save your changes by clicking **Apply** or **OK**, and then select **Save**.

The migration tools convert appropriate command-line parameters to Java virtual machine settings in the process server definition. Most settings are mapped directly, but some settings are not migrated because their roles differ in WebSphere Application Server version 7.0.x. In such cases, the configuration settings might not exist, they might have different meanings, or they might have different scopes. See the following topics in the WebSphere Application Server version 7.0.x information center for more information about changing the process definition settings or JVM settings:

- Process definition settings

- Java virtual machine settings

Related tasks:

“Migrating the security configuration for a stand-alone environment” on page 358
You can migrate the security configuration and security settings from a previous version of WebSphere ESB to the current version.

“Migrating the security configuration for a network deployment environment” on page 358
You can migrate the security configuration and security settings from a previous version of WebSphere ESB to the current version.

Runtime migration tools reference

Use the runtime migration tools to migrate topology configuration, applications, and databases to WebSphere ESB V7.5.1.

The runtime migration tools required to perform a version-to-version migration fall into the following categories:

“WebSphere ESB profile migration wizard”

“WebSphere ESB migration command-line utilities”

“WebSphere Application Server migration command-line utilities” on page 370

WebSphere ESB profile migration wizard

The WebSphere ESB profile migration wizard is a graphical user interface (GUI) that guides you through the process of migrating a profile. The wizard is invoked by running the **BPMigrate** command.

For more information about the **BPMigrate** command, see the BPMigrate command topic.

For more information about running the WebSphere ESB profile migration wizard, see Migrating a profile using the WebSphere ESB profile migration wizard.

WebSphere ESB migration command-line utilities

The WebSphere ESB migration command-line utilities are located in the `INSTALL_ROOT/bin` directory, except for **BPMigrate**, which is located in the `INSTALL_ROOT/bin/bpm_migration` directory.

BPMCreateDatabaseUpgradeUtilities

The **BPMCreateDatabaseUpgradeUtilities** command creates an archive file that contains all of the commands and their prerequisites that need to be invoked on the database system where the database upgrade is to be performed.

For more information about the **BPMCreateDatabaseUpgradeUtilities** command, see the BPMCreateDatabaseUpgradeUtilities command-line utility topic.

BPMCreateRemoteMigrationUtilities

The **BPMCreateRemoteMigrationUtilities** command creates an archive file containing all the commands and their prerequisites that need to be invoked on the system containing the source profile to be migrated.

For more information about the **BPMCreateRemoteMigrationUtilities** command, see the **BPMCreateRemoteMigrationUtilities** command-line utility topic.

BPMCreateTargetProfile

The **BPMCreateTargetProfile** command creates a target migration profile using some of the base configuration information that was backed up using the **BPMSnapshotSourceProfile** command.

For more information about the **BPMCreateTargetProfile** command, see the **BPMCreateTargetProfile** command-line utility topic.

BPMGenerateUpgradeSchemaScripts

The **BPMGenerateUpgradeSchemaScripts** command generates SQL scripts and upgradeSchema scripts for a database that must be upgraded manually.

For more information about the **BPMGenerateUpgradeSchemaScripts** command, see the **BPMGenerateUpgradeSchemaScripts** command-line utility topic.

BPMMigrate

The **BPMMigrate** command invokes the Business Process Management profile migration wizard that supports the migration of Business Process Management profiles.

For more information about the **BPMMigrate** command, see the **BPMMigrate** command-line utility topic.

For more information about running the WebSphere ESB profile migration wizard, see “Migrating a profile using the profile migration wizard” on page 345.

BPMMigrateCluster

The **BPMMigrateCluster** command migrates cluster-scoped application and configuration information.

For more information about the **BPMMigrateCluster** command, see the **BPMMigrateCluster** command-line utility topic.

BPMMigrateProfile

The **BPMMigrateProfile** command migrates a source profile from the snapshot directory to a target profile.

For more information about the **BPMMigrateProfile** command, see the **BPMMigrateProfile** command-line utility topic.

BPMMigrationStatus

The **BPMMigrationStatus** command displays the status of the migrations that have been executed on the system.

For more information about the **BPMMigrationStatus** command, see the **BPMMigrationStatus** command-line utility topic.

BPMQueryDeploymentConfiguration

The **BPMQueryDeploymentConfiguration** command extracts the deployment configuration from the source profile and generates an XML file. This information is needed specifically for updating WebSphere Adapters during runtime migration.

For more information about the **BPMQueryDeploymentConfiguration** command, see the **BPMQueryDeploymentConfiguration** command-line utility topic.

BPMSnapshotSourceProfile

The **BPMSnapshotSourceProfile** command copies the configuration files in the source profile to a snapshot directory that will serve as the source of the profile migration.

For more information about the **BPMSnapshotSourceProfile** command, see the **BPMSnapshotSourceProfile** command-line utility topic.

migrateBSpaceData (Business Space)

Use the **migrateBSpaceData** command-line utility to migrate the Business Space data.

For more information about the **migrateBSpaceData** command, see the **migrateBSpaceData** command-line utility topic.

upgradeSchema (Common Database)

Use the **upgradeSchema** command-line utility to upgrade the database schema.

For more information about the **upgradeSchema** command, see the **upgradeSchema** topic.

WebSphere Application Server migration command-line utilities

backupConfig

The **backupConfig** command is a simple utility to back up the configuration of your node to a file.

For more information about the **backupConfig** command, see the **backupConfig** command topic on the WebSphere Application Server information center.

convertScriptCompatibility

The **convertScriptCompatibility** command is used by administrators to convert their configurations from a mode that supports backward compatibility of WebSphere Application Server Version 5.1.x or Version 6.0.x administration scripts to a mode that is fully in the Version 7.0 configuration model.

For more information about the **convertScriptCompatibility** command, see the **convertScriptCompatibility** command topic on the WebSphere Application Server information center.

migrationDisablementReversal

If you need to roll back a deployment cell or managed node, use the **wsadmin** command to run the **migrationDisablementReversal.jacl** script.

For more information about the **migrationDisablementReversal.jacl** script, see the **Rolling back a Network Deployment cell** topic on the WebSphere Application Server information center.

restoreConfig

Use the **restoreConfig** command to restore the configuration of your node after backing up the configuration using the **backupConfig** command.

For more information about the **restoreConfig** command, see the **restoreConfig** command topic on the WebSphere Application Server information center.

startManager

Use the **startManager** command to manipulate a deployment manager with scripting.

For more information about the **startManager** command, see the **startManager** command topic on the WebSphere Application Server information center.

startNode

The **startNode** command reads the configuration file for the node agent process and constructs a launch command.

For more information about the **startNode** command, see the **startNode** command topic on the WebSphere Application Server information center.

startServer

The **startServer** command reads the configuration file for the specified server process and starts that server process.

For more information about the **startServer** command, see the **startServer** command topic on the WebSphere Application Server information center.

stopManager

The **stopManager** command reads the configuration file for the Network Deployment manager process.

For more information about the **stopManager** command, see the **stopManager** command topic on the WebSphere Application Server information center.

stopNode

The **stopNode** command reads the configuration file for the Network Deployment node agent process and sends a Java Management Extensions (JMX) command telling the node agent to shut down.

For more information about the **stopNode** command, see the **stopNode** command topic on the WebSphere Application Server information center.

stopServer

The **stopServer** command reads the configuration file for the specified server process. This command sends a Java management extensions (JMX) command to the server telling it to shut down.

For more information about the **stopServer** command, see the **stopServer** command topic on the WebSphere Application Server information center.

syncNode

The **syncNode** command forces a configuration synchronization to occur between the node and the deployment manager for the cell in which the node is configured.

The node agent server runs a configuration synchronization service that keeps the node configuration synchronized with the master cell configuration. If the node agent is unable to run because of a problem in the node configuration, you can use the **syncNode** command to perform a

synchronization when the node agent is not running in order to force the node configuration back in sync with the cell configuration. If the node agent is running and you want to run the **syncNode** command, you must first stop the node agent.

For more information on the **syncNode** command, see the **syncNode** command topic on the WebSphere Application Server information center.

Runtime migration troubleshooting

Review this page for troubleshooting tips if you encounter problems while migrating from an earlier version of WebSphere ESB.

The following sections describe specific errors and exceptions that might occur in a runtime version migration of WebSphere ESB and provide steps you can follow to understand and resolve these problems.

Errors that result from migration command failures are logged in the snapshot directory in files whose names begin with *commandname* and end with *.error*. You can use these files to determine the exact place the problem occurred.

- “Application installation error”
- “Application server error” on page 373
- “Business Space migration exception” on page 373
- “Communication with deployment manager error” on page 374
- “ConnectorException” on page 374
- “Exceptions: database connectivity, loading, or missing class” on page 374
- “LifecycleServiceError” on page 375
- “Missing step BPMMigrateCluster” on page 375
- “Out of memory error” on page 375
- “Profile creation error” on page 376
- “Profile migration error” on page 376
- “Warning: There are date after end of archive” on page 378
- “WebSphere ESB client migrations” on page 378
- “WebSphere ESB client migrations” on page 378
- “WSDL validation exception” on page 378

Application installation error

If you select the option for the migration process to install the enterprise applications that exist in the V7.5, 7.0, 6.2.0, or 6.1.0 configuration into the new WebSphere ESB V7.5.1 configuration, you might encounter some error messages during the application-installation phase of migration.

The applications that exist in the WebSphere ESB V7.5, 7.0, 6.2.0, or 6.1.0 configuration might have incorrect deployment information—usually, incorrect XML documents that were not validated sufficiently in previous WebSphere ESB runtime environments. The runtime environment now has an improved application-installation validation process and will fail to install these malformed EAR files. This results in a failure during the application-installation phase of **BPMigrateProfile** and produces an “E:” error message.

If the application installation fails in this way during migration, you can do one of the following:

- Fix the problems in the V7.5, 7.0, 6.2.0, or 6.1.0 applications, and then remigrate.
- Proceed with the migration and ignore these errors.

In this case, the migration process does not install the failing applications but does complete all of the other migration steps.

Later, you can fix the problems in the applications and then manually install them in the new V7.5.1 configuration using the administrative console or an installation script.

Application server error

After you migrate a managed node to V7.5.1, the application server might not start.

When you try to start the application server, you might see errors similar to those in the following example:

```
[5/11/06 15:41:23:190 CDT] 0000000a SystemErr R
    com.ibm.ws.exception.RuntimeError:
com.ibm.ws.exception.RuntimeError: org.omg.CORBA.INTERNAL:
    CREATE_LISTENER_FAILED_4
vmcid: 0x49421000 minor code: 56 completed: No
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr R at
com.ibm.ws.runtime.WsServerImpl.bootServerContainer(WsServerImpl.java:198)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr R at
com.ibm.ws.runtime.WsServerImpl.start(WsServerImpl.java:139)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr R at
com.ibm.ws.runtime.WsServerImpl.main(WsServerImpl.java:460)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr R at
com.ibm.ws.runtime.WsServer.main(WsServer.java:59)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr R at
sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
[5/11/06 15:41:23:196 CDT] 0000000a SystemErr R at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:64)
[5/11/06 15:41:23:197 CDT] 0000000a SystemErr R at
sun.reflect.DelegatingMethodAccessorImpl.invoke
    (DelegatingMethodAccessorImpl.java:43)
```

Change the port number at which the managed node's server is listening. If the deployment manager is listening at port 9101 for ORB_LISTENER_ADDRESS, for example, the server of the managed node should not be listening at port 9101 for its ORB_LISTENER_ADDRESS. To resolve the problem in this example, perform the following steps:

1. On the administrative console, click **Application servers > server_name > Ports > ORB_LISTENER_ADDRESS**.
2. Change the ORB_LISTENER_ADDRESS port number to one that is not used.

Business Space migration exception

By default, the Business Space server attempts to read every user attribute from the user Lightweight Directory Access Protocol (LDAP) repository; however, the WebSphere Application Server WAP Identity Module (WIM) component has trouble resolving some of the attribute fields that point to a remove entity in the user repository. If you see errors in the profile_root/logs/server_name/systemout.log file after migration similar to
com.ibm.websphere.wim.exception.WIMSystemException: CWWIM1013E The value of

the property secretary is not valid for entity, follow steps 1 and 2 to limit the number of user attributes that are required for Business Space to run successfully.

1. In the administrative console, navigate to: **Resources > Resource Environment > Resource environment providers > Mashups_ConfigService > Custom properties.**
2. Modify or add the following property and set the value to LIMITED:
Name - com.ibm.mashups.user.userProfile
Value - LIMITED

Communication with deployment manager error

Sometimes the migration process can fail because of insufficient resources on the machine. If the migration fails, check the log file to see if the following message appears:

```
"MIGR0494E: An unexpected error occurred during communication with the Deployment Manager, the migration cannot continue. Resolve the error and rerun the WASPreUpgrade tool to create a new backup directory."
```

If you see this message in the log file, check the disk space on the machine, memory and CPU utilization. If possible, stop some other processes on the machine to free up machine resources and rerun the migration command that has failed.

ConnectorException

When migrating a managed node, if you see a ConnectorException as follows, ensure that your deployment manager is running and rerun the command.

```
MIGR0380E: The JMX connection is not established with the deployment manager node qaxs06, using connector type of SOAP on port 8879. The WASPostMigration program is now closing. No changes are made to the local Application Server environment.  
com.ibm.websphere.management.exception.ConnectorException: ADMC0016E: The system cannot create a SOAP connector to connect to host qaxs06 at port 8879.  
com.ibm.ws.migration.utility.UpgradeException:  
com.ibm.websphere.management.exception.ConnectorException: ADMC0016E: The system cannot create a SOAP connector to connect to host qaxs06 at port 8879.
```

Exceptions: database connectivity, loading, or missing class

Never change any WebSphere Application Server variables that are configured as a part of profile creation.

If you modify these values incorrectly in old profile, you might get database connectivity, loading, or other missing class exceptions, such as:

```
10/25/08 13:22:39:650 GMT+08:00] 0000002e J2CUtilityCla E J2CA0036E: An exception occurred while invoking method setDataSourceProperties on com.ibm.ws.rsadapter.spi.WSManagedConnectionFactoryImpl used by resource jdbc/com.ibm.ws.sib/ewps6101.Messaging-BPC.cwfpccell01.Bus :  
com.ibm.ws.exception.WsException: DSRA0023E: The DataSource implementation class "com.ibm.db2.jcc.DB2XADDataSource" could not be found.DB2,
```

SQL Embedded JDBC drivers are bundled with the WebSphere ESB product installation. If you need to change these drivers to any higher version, you must copy drivers on the same location where they exist in the product installation, as follows:

- **DB2:**`%was.install.root%/universalDriver_wbi/lib`

- `SQL:%was.install.root%lib`

If you need a new JDBC provider and datasource for your application, you can create these resources by selecting a valid `jdbcclasspath` and setting the WebSphere Application Server variable accordingly. For example, if you need DB2 at cell level which doesn't exist earlier in your installation, you could use the following procedure.

1. In the administrative console, navigate to: **Resources > JDBC > JDBC Providers > DB2 Universal JDBC Driver Provider (XA)**.
2. In the **Class path** box, set the following paths:
 - `DB2UNIVERSAL_JDBC_DRIVER_PATH = %was.install.root%/universalDriver_wbi/lib`
 - `DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH=""`

If you need your own drivers, set the following path:
`DB2UNIVERSAL_JDBC_DRIVER_PATH=%myDriverLocation%`

LifeCycleServiceError

The **LifeCycleService** error can occur as a postmigration exception if changesets in the database repository are in the active state.

This error applies only to a version 6.2 to 7.x migration scenario.

You can avoid this error by checking for changes before migrating WebSphere ESB version 6.2 to 7.x, as described in Runtime premigration checklist.

Missing step BPMigrateCluster

If the network deployment migration process misses the step **BPMigrateCluster** for a cluster, the servers in that cluster show the following error message in the server log file:

```
[3/4/10 10:52:20:767 CST] 00000000 WsServerImpl E WSVR00096E: Error occurred during startup
com.ibm.ws.exception.RuntimeError: BPM00996E: The cluster <cluster name> has not been migrated to version that's compatible with node suse0Node01. Please execute BPMigrateCluster command on the deployment manager profile to complete cluster migration.
at com.ibm.bpm.migration.cluster.detection.DetectClusterMigration.start(DetectClusterMigration.java:135)
at com.ibm.ws.runtime.component.ContainerHelper.startComponents(ContainerHelper.java:538)
at com.ibm.ws.runtime.component.ContainerImpl.startComponents(ContainerImpl.java:167)
```

Perform the following actions to correct this error:

1. Execute the **BPMigrateCluster** command on the deployment manager profile to migrate the cluster configuration.
2. Perform a `syncNode` operation for all managed nodes that participate in the cluster.
3. Restart the cluster.

Out of memory error

If either the **BPMSnapshotSourceProfile** or **BPMigrateProfile** command-line utility fails due to Out of Memory problems, you can increase the heap size to a number that takes into consideration the size and scope of the environment being migrated, as well as what the machine will allow.

For instructions on how to increase the heap size, use the procedure described in Solution 4 of the following technote: Handling certain Out of Memory conditions when migrating an earlier version of WebSphere Application Server to V6.0.2, V6.1, or 7.0.

Profile creation error

While you are using the V7.5.1 migration wizard to create a profile when migrating a configuration, you might see the following profile-creation error messages.

```
profileName: profileName cannot be empty
profilePath: Insufficient disk space
```

These error messages might be displayed if you enter a profile name that contains an incorrect character such as a space. Rerun the migration wizard, and verify that there are no incorrect characters in the profile name such as a space, quotation marks, or any other special characters.

Profile migration error

When you use the migration wizard to migrate a profile from WebSphere ESB V7.5, 7.0, 6.2.0, or 6.1.0 to WebSphere ESBV7.5.1 on a Solaris x64 processor-based system, the migration might fail during the **BPMigrateProfile** step.

You might see messages similar to the following in *profile_root/logs/WASPostUpgrade.time_stamp.log*:

```
MIGR0327E: A failure occurred with stopNode.
MIGR0272E: The migration function cannot complete the command.
```

WebSphere ESB V7.5, 7.0, 6.2.0, or 6.1.0 uses a Java virtual machine (JVM) in 32-bit mode. The migration wizard for WebSphere ESB V7.5.1 calls the **BPMigrateProfile.sh** script, which attempts to run the JVM for V7.5, 7.0, 6.2.0, or 6.1.0 in the 64-bit mode when the server stops the V7.5, 7.0, 6.2.0, or 6.1.0 node.

Complete the following actions to remove the incomplete profile and enable WebSphere ESB to correctly migrate the V7.5, 7.0, 6.2.0, or 6.1.0 profile:

1. On a command line, change to the *install_root/bin* directory.
For example, type the following command:

```
cd /opt/IBM/WebSphere/ESB/bin
```
2. Locate the **BPMigrateProfile.sh** script in the *install_root/bin* directory, and make a backup copy.
3. Open the **BPMigrateProfile.sh** or **BPMigrateProfile.bat** file in an editor, and perform the following actions:
 - a. Locate the following line of code:

```
UNIX      Linux
"$binDir" /setupCmdLine.sh
```

```
Windows
call "%dp0setupCmdLine.bat" %*
```
 - b. Insert the following line of code after the code that was identified in the previous step:

```
JVM_EXTRA_CMD_ARGS=""
```
 - c. Save the changes.
4. Repeat steps 2 through 4 with the **WASPostUpgrade.sh** or the **WASPostUpgrade.bat** file.
5. Delete the incomplete V7.5.1 profile that was created during the migration process. Use the following procedure.

- a. Open a command prompt and run one of the following commands, based on your operating system:

- **Linux** **UNIX** **On Linux and UNIX platforms:** `manageprofiles.sh -delete -profileName profile_name`
- **Windows** **On Windows platforms:** `manageprofiles.bat -delete -profileName profile_name`

The variable *profile_name* represents the name of the profile that you want to delete.

- b. Confirm that the profile deletion has completed by checking the following log file:

- **Linux** **UNIX** `install_root/logs/manageprofiles/
profile_name_delete.log`
- **Windows** `install_root\logs\manageprofiles\profile_name_delete.log`

6. Delete the *profile_root* directory of the V7.5.1 profile that was removed in the previous step.

7. Rerun the migration wizard.

Synchronization error

If synchronization fails when you migrate a managed node to V7.5.1, the server might not start.

You might receive messages similar to the following when you migrate a managed node to V7.5.1:

```
ADMU0016I: Synchronizing configuration between node and cell.
ADMU0111E: Program exiting with error:
           com.ibm.websphere.management.exception.AdminException: ADMU0005E:
           Error synchronizing repositories
ADMU0211I: Error details may be seen in the file:
           /opt/WebSphere/62AppServer/profiles/AppSrv02/logs/syncNode.log
MIGR0350W: Synchronization with the deployment manager using the SOAP protocol
           failed.
MIGR0307I: The restoration of the previous WebSphere Application Server
           environment is complete.
MIGR0271W: Migration completed successfully, with one or more warnings.
```

These messages indicate the following:

- Your deployment manager is at a V7.5.1 configuration level.
- The managed node that you are trying to migrate is at a V7.5.1 configuration level on the deployment manager's repository (including applications).
- The managed node is not quite complete because you did not complete the syncNode operation.

Perform the following actions to resolve this issue:

1. Rerun the **syncNode** command on the node to synchronize it with the deployment manager.
See the syncNode command .
2. Run the **GenPluginCfg** command.
See the GenPluginCfg command .

Warning: There are date after end of archive

If the **BPMCreateDatabaseUpgradeUtilities** command is executed on a UNIX platform but the generated .zip file would be used on a Windows platform for execution of the **BPMGenerateUpgradeSchemaScripts** command, during extraction of the .zip file you might see the following warning: Warning: There are date after end of archive. This warning is harmless, and the extracted content and command should work successfully.

WebSphere ESB client migrations

When migrating WebSphere ESB client profiles from source V7.5, 7.0, 6.2.0, or 6.1.0 to a full server WebSphere ESB V7.5.1 installation, the target profile augmentation is not correct. Applications on the target profile might not work correctly. To correct the problem, use the **manageprofiles** command-line utility to add the augmentation for `INSTALL_ROOT/profileTemplates/SCA/*.sdo` template, where the "*" symbol represents "default" for standalone and "managed" for federated profiles.

WSDL validation exception

If the **BPMmigrateProfile** command fails with the following WSDL validation exception, it means that a WSDL file in the application that failed to install has an input element declaration that is not defined within an operation. To fix this problem, you must either define the input element declaration or remove it from the WSDL file.

WSDL validation exception

```
java.io.IOException: javax.wsdl.WSDLException: WSDLException (at /wsdl:definitions/wsdl:import/wsdl:definitions/wsdl:input): faultCode=INVALID_WSDL: Encountered illegal extension element '{http://schemas.xmlsoap.org/wsdl/}input' in the context of a 'javax.wsdl.Definition'. Extension elements must be in a namespace other than WSDL's.
javax.wsdl.WSDLException: WSDLException (at /wsdl:definitions/wsdl:import/wsdl:definitions/wsdl:input): faultCode=INVALID_WSDL: Encountered illegal extension element '{http://schemas.xmlsoap.org/wsdl/}input' in the context of a 'javax.wsdl.Definition'. Extension elements must be in a namespace other than WSDL's.
```

How to fix the problem

Use the following procedure to fix the problem.

1. Locate the WSDL file in the application that failed to install. The WSDL file that is failing in validation has an input element declaration that is not defined within an operation.

Sample of a failed WSDL file:

Note: The declaration for `getLastSellPriceRequest` is not defined under the `wsdl:operation` declaration.

```
wsdl:portType name="EnrollIntf"
wsdl:operation name="Enrollment"
wsdl:input message="tns:EnrollmentRequestMsg" name="EnrollmentRequest"/
wsdl:output message="tns:EnrollmentResponseMsg" name="EnrollmentResponse"/
/wsdl:operation
/wsdl:portType

wsdl:input name="getLastSellPriceRequest"
wsdl:soap:header message="tns:EnrollmentRequest" part="soap_header" use="literal"/
wsdl:soap:body parts="EnrollReq" use="literal"/
/wsdl:input
```

2. Make the appropriate change to the input declaration, depending on whether the input declaration file is needed or not.

- If the input declaration is needed, move it under the operation that uses it.
 - If the input declaration is not needed, remove it from the WSDL file.
3. Update the application in source environment.
 4. Verify that the application works in the source environment.
 5. Perform the migration steps again, starting with the **BPMSnapshotSourceProfile** command or the WebSphere ESB profile migration wizard.

[Back to the top](#)

WebSphere ESB deprecated and removed features

This section details features deprecated or removed in WebSphere ESB. Deprecated features from other WebSphere Application Server product offerings are described in the documentation for those products. WebSphere ESB version 6.1 was the first release to have any deprecation.

WebSphere ESB deprecation list

This topic describes the deprecated or removed features in WebSphere ESB v6.1 and later versions and releases.

The following tables summarize what is deprecated, by version and release. Each table reflects the version and release where the deprecation took effect and lists what is being deprecated, such as features, APIs, scripting interfaces, tools, wizards, publicly exposed configuration data, naming identifiers, and constants. Where possible, a recommended migration action is provided.

Deprecated features in WebSphere ESB v7.5.1

The following items are deprecated in WebSphere ESB 7.5.1 from previous releases:

| Deprecation | Description | Recommended action |
|-----------------|---|---|
| Browser support | Support for the browsers Microsoft Internet Explorer Version 6, Microsoft Internet Explorer Version 7, and Mozilla Firefox Version 3.6 is deprecated. | Use Microsoft Internet Explorer Version 8, or later, or use Mozilla Firefox version 4.0 or later. |

Deprecated and removed features in WebSphere ESB v7.5

The following items are deprecated in WebSphere ESB 7.5 from previous releases:

| Deprecation | Description | Recommended action |
|--|---|---|
| Configuring a deployment environment while creating a deployment manager or custom profile using the Profile Management tool and manageprofiles command line utility | The option to configure a deployment environment while creating a deployment manager or custom profile is deprecated in WebSphere ESB V7.5. | Use the administration console wizard or equivalent AdminTask APIs. |

| Deprecation | Description | Recommended action |
|---|--|---|
| Configuring a custom deployment environment using the Deployment Environment wizard | The ability to configure a custom deployment environment using the Deployment Environment wizard is removed in WebSphere ESB V7.5. | To configure a custom deployment environment, configure each application server cluster (and product component) individually using their associated console panels. |

Deprecated features in WebSphere ESB v7.0

| IBM Web Services Clients |
|--|
| IBM Web Services Clients is no longer packaged with Websphere ESB. For information on obtaining Web Services Clients refer to the Message Service Clients section in the Overview. |

Deprecated features in WebSphere ESB v6.2

| |
|---|
| There are no deprecated features for WebSphere ESB v6.2 |
| |

Deprecated features in WebSphere ESB v6.1.2

| |
|---|
| There are no deprecated features for WebSphere ESB v6.1.2 |
| |

Deprecated features in WebSphere ESB v6.1

| IBM Web Services Client for C++ |
|---|
| <p>The WebSphere ESB and WebSphere Process Server products do not use or have a dependency on this feature. The feature was provided as an independent tool which could be used in one of two ways; either as a prerequisite for the IBM Message Service Client for C/C++, or, as a web services toolkit for C++ applications.</p> <p>Recommended migration action:</p> <p>If you have an application which uses this tool in either of the possible uses the following steps should be taken.</p> <ol style="list-style-type: none"> 1. If used as a prerequisite for the IBM Message Service Client for C/C++. This dependency is no longer required for the IBM Message Service Client for C/C++. If a previous version of the IBM Message Service Client for C/C++ is installed it should be refreshed with the latest version available from the installation media. Any existing version of the IBM Web Services Client for C++ can then be safely removed. 2. If used as a web services toolkit for C++ applications. This functionality is no longer supported and must be replicated by using an alternative external tool. Several choices are available, including, for example, gSOAP, which is an open source product available from http://www.cs.fsu.edu/~engelen/soap.html. |

Index

A

- adapter 307, 309, 345, 349, 351, 353
- administrative console
 - network deployment configuration 228
- augmentation 302
- authentication 358
 - databases 15
- authorization 15, 358

B

- backupconfig 317, 322, 331, 361, 364, 368
- bus 39, 290, 302
- business process rules manager 307
- business rule 307, 322, 331
- business space 317, 322, 331, 356, 372

C

- CEI
 - database 244
 - databases 55
- cell 10, 303, 307, 322, 331, 361, 368
- client 379
- cluster 15, 303, 307, 322, 331, 345, 349, 351, 353, 356, 372
- CMNDB database
 - planning 49
- command 300, 317, 322, 331, 349, 361, 364, 368, 372
- common database 317, 331, 356
 - planning 49
- component 15
- components
 - configurations 33
 - creating database design files for 102, 211
- configuration
 - network deployment 228
 - wsadmin 150
 - roadmap
 - WebSphere ESB 2
 - widgets
 - WebSphere Portal 244
- configurations
 - topology patterns 33
- configuring
 - WebSphere ESB 111
- connectivity 372
- custom profiles
 - creating
 - Profile Management Tool 140
- customized topologies 31
- customized topology 31
 - compared to Remote Messaging and Remote Support topology 31

- customized topology (*continued*)
 - Remote Messaging and Remote Support topology
 - compared to customized topology 31

D

- data source 39, 307, 309, 317, 322, 331
- data store 322, 331
- database
 - CEI 55, 244
- database administrators
 - tasks 41
- database configurations 59
 - planning messaging engines 55
- database connectivity 372
- database design files
 - creating 97, 102, 206, 211
- database design tool
 - creating database design files 97, 206
 - creating database design files using 97, 102, 206, 211
 - troubleshooting 105, 215
- database schema 317, 322, 331, 356
- databases 51
 - JDBC drivers 51
 - naming restrictions 39
 - overview 35
 - planning 35
 - product components 49
 - security 15
 - SQL scripts 51
 - supported types 51
 - topology considerations 35
 - user ID requirements 51
 - user IDs 15
- deployment cell 361, 368
- deployment environment 290, 315, 322, 331, 345
 - creating database design files for 97, 206
- deployment manager 10, 302, 322, 331, 372
- deprecated feature 289, 379
- disk space
 - overview 8
 - planning 8
- drivers 40

E

- enterprise service bus 39
- environments
 - network 18
 - stand-alone 18
- EVENT database
 - planning 49

H

- hardware
 - planning 8

I

- IBMWSSIB schema
 - messaging engine 55
- installation
 - roadmap
 - WebSphere ESB 2
 - verification 90
- installing
 - interim fix 283
 - silently 284
 - silent 85
 - command line 85
 - response file 88
 - updates 281
 - interactively 281
 - WebSphere ESB 1
- interim fix
 - installing 283
 - silently 284
- uninstalling
 - silently 285

J

- jacl 361, 364
- jdbc 39, 304, 307, 309, 317, 322, 331, 372
- JDBC 40
- JDBC drivers 40
 - databases 51

L

- logger mediation
 - database configurations 59

M

- managed node 322, 331, 364, 372
- manageprofiles 345, 349, 351, 372
- manageProfiles
 - examples 154
- manual migration 304
- MEDB database
 - messaging engine 55
- messaging 39, 317, 322, 331
- messaging engine 317, 322, 331
 - database configurations 55
- migration 289, 290, 291, 293, 295, 298, 299, 300, 302, 303, 304, 306, 307, 308, 309, 315, 317, 322, 331, 345, 349, 351, 353, 356, 359, 361, 368, 372

N

- network deployment 290, 299, 315, 322, 331, 345
 - configuration
 - wsadmin 150
 - topology patterns 25
- network deployment configuration
 - custom installation 228
- network environments
 - choosing 18
- node 10, 309, 322, 331, 345, 361, 364, 372
- node agent 322, 331
- nonadministrative users
 - installation 42

O

- OBSRVDB database
 - planning 49
- oracle 304, 309, 317, 322, 331
- overview
 - topology patterns 33

P

- port 345, 372
- Portal
 - configuration
 - widgets 244
- profile 10, 299, 300, 302, 307, 309, 317, 322, 331, 345, 349, 351, 353, 368, 372
- Profile Management Tool
 - custom profiles
 - creating 140
- profiles
 - augmenting 174
 - custom
 - Profile Management Tool 140
 - database considerations 51

R

- Remote Messaging and Remote Support topology pattern 28
 - compared to Remote Messaging topology pattern 28
 - overview 21
- Remote Messaging topology pattern
 - compared to Remote Messaging and Remote Support topology pattern 28
- Remote Messaging topology pattern 27
 - compared to Single Cluster topology pattern 27
 - overview 21
- Single Cluster topology pattern
 - compared to Remote Messaging topology pattern 27
- Remote Messaging, Remote Support, and Web topology pattern
 - overview 21
- Remote Messaging, Support and Web Applications topology pattern 30

- Remote Messaging, Support and Web Applications topology pattern
 - (continued)
 - compared to Remote Messaging and Remote Support topology pattern 30
- Remote Messaging and Remote Support topology pattern
 - compared to Remote Messaging, Support and Web Applications topology pattern 30
- removing
 - interactively 286
 - silent
 - uninstall 287
 - silently 287
 - software 286
- requirements
 - overview 8
 - planning 8
 - software versions 9
- resources
 - overview 8
 - planning 8
 - software versions 9
- restoreconfig 361, 364, 368
- roadmap
 - installation
 - WebSphere ESB 2
- rolling back
 - updates 282
- rules manager 307, 322

S

- SCADB database
 - messaging engine 55
 - schema 317, 322, 331, 356
 - security 15, 317, 322, 331
 - databases 15
 - service component 15
 - service component architecture 15
 - silent
 - installation 85
 - command line 85
 - response file 88
- Single Cluster topology pattern 25
 - compared to Remote Messaging topology pattern 25
 - overview 21
 - Remote Messaging topology pattern
 - compared to Single Cluster topology pattern 25
- SQL scripts
 - databases 51
- stand-alone environments
 - choosing 18
- stand-alone profile 317, 351
- stand-alone profiles
 - creating database design files for 97, 206
- stand-alone server 345, 353
- system requirements
 - overview 8
 - planning 8
 - software versions 9

T

- table space 322, 331
- topologies
 - customized 31
- topology patterns
 - considerations 32
 - network deployment 25
 - overview 33
 - Remote Messaging 21, 27
 - Remote Messaging and Remote Support 21, 28
 - Remote Messaging, Remote Support, and Web 21
 - Remote Messaging, Support and Web Applications 30
 - Single Cluster 21, 25
- troubleshooting
 - database design tool 105, 215

U

- uninstalling
 - interactively 286
 - interim fix
 - silently 285
 - silently 287
 - software 286
- updates
 - installation 281
 - rolling back 282
- updating
 - software 281
- user IDs
 - databases 15, 51

V

- verification
 - installation 90
- version-to-version migration 290, 291

W

- wsadmin 345, 358, 361, 364
 - network deployment
 - configuration 150
- wsdl 372

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Software Interoperability Coordinator, Department 49XA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software for use with this program.

This book contains information on intended programming interfaces that allow the customer to write programs to obtain the services of WebSphere ESB.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Important: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks

IBM, the IBM logo, ibm.com[®], are trademarks of IBM Corporation, registered in many jurisdictions worldwide. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” www.ibm.com/legal/copytrade.shtml. Other product and service names might be trademarks of IBM or other companies.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

This product includes software developed by the Eclipse Project (<http://www.eclipse.org/>).

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Sending your comments to IBM

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM.

Feel free to comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness of this book.

Please limit your comments to the information in this book and the way in which the information is presented.

To make comments about the functions of IBM products or systems, talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate, without incurring any obligation to you.

You can send your comments to IBM in any of the following ways:

- By mail, to this address:

User Technologies Department (MP095)
IBM United Kingdom Laboratories
Hursley Park
WINCHESTER,
Hampshire
SO21 2JN
United Kingdom

- By fax:
 - From outside the U.K., after your international access code use 44-1962-816151
 - From within the U.K., use 01962-816151
- Electronically, use the appropriate network ID:
 - IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
 - IBMLink: HURSLEY(IDRCF)
 - Internet: idrcf@hursley.ibm.com

Whichever method you use, ensure that you include:

- The publication title and order number
- The topic to which your comment applies
- Your name and address/telephone number/fax number/network ID.



GC34-7235-01

