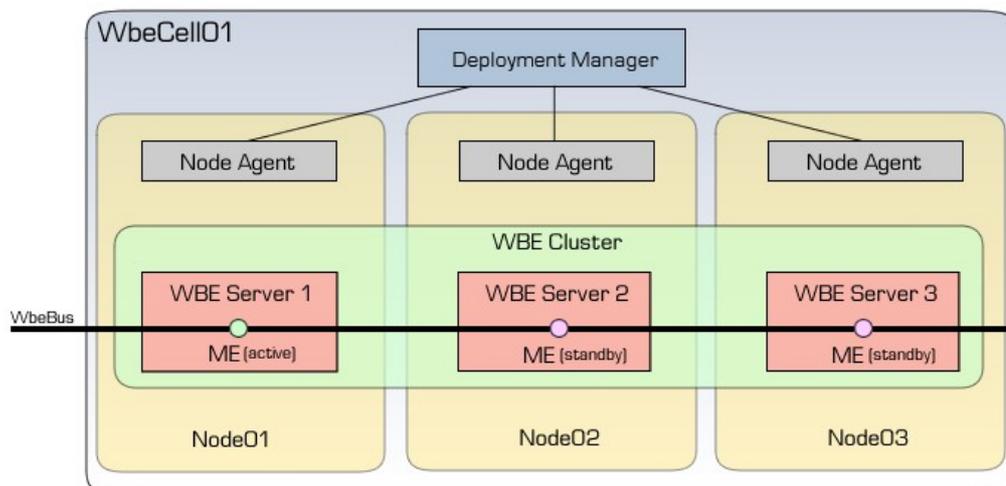


Installing WebSphere Business Events 6.2.0.1 into the Silver Cluster Topology

The Silver Topology Cluster Environment

The silver cluster topology consists of a single WebSphere Application Server (WAS) cluster and a single active messaging engine. This topology allows for high availability because the processing of events occurs on all of the active servers, and failure of a server within the cluster results in failover of any active messaging without disrupting the event processing.

A silver topology cluster with three nodes and three servers hosting the WebSphere Business Events (WBE) runtime is illustrated in the following diagram:



The golden topology, however, enables scalability for both high availability and performance; because the silver topology has only one active messaging engine, all pre-processing and distribution of events are performed by a single cluster member. Consider using the golden topology rather than the silver topology if possible.

The following instructions create a silver topology cluster with a DB2 repository and the WebSphere default messaging provider (SIBus). Modification is required if you are attempting to configure a cluster that differs from this topology or that uses different databases or messaging technologies.

Installing WebSphere Business Events

On each machine that will be part of the cluster, perform the following installation steps to install WebSphere Application Server and WebSphere Business Events:

1. Install WebSphere Application Server 6.1 Network Deployment.
2. Install WebSphere Update Installer.
3. Install WebSphere Application Server Fix Pack 21 or later.
4. Install the Feature Pack for Web Services for WebSphere Application Server.
5. Install Feature Pack for Web Services for WebSphere Application Server Fix Pack 21 or higher.
6. Install WebSphere Business Events 6.2 and select the Advanced Installation option.
7. Install the WebSphere Business Events 6.2 Fix Pack 1.
8. Install Object Grid:
 1. In a command window (Windows) or terminal (Linux and UNIX), change to the directory called og in the location where WebSphere Business Events is installed.
 2. Run the following command:

```
java -cp og_install.jar run -OPT launchOption="oginstall"
```

3. The ObjectGrid installer starts.
4. When prompted, specify the installation location of the WebSphere Application Server instance that you want to use with WebSphere Business Events. The installer identifies the profiles that exist in that instance of WebSphere Application Server.
5. When prompted, specify which profiles you want to update with ObjectGrid. You must update any profile on which you intend to run WebSphere Business Events.
6. If you are prompted by the installer, stop any profiles that are running before they can be updated.
9. Start the profile management tool from
was_install_dir/bin/ProfileManagement/pmt.bin
10. Create a new stand-alone application server profile (**Create > Feature Pack for Web Services > Application server with Feature Pack for Web Services**)

Set up your clusters

This step will create the additional servers and set up the clustered environment, into which WebSphere Business Events will be deployed.

Create a deployment manager

On one of the machines on which you installed WAS and WBE, create a deployment manager (these steps are run only on a single machine):

1. From *was_install_dir/bin/ProfileManagement*, run the pmt command to launch the Profile Management Tool.

2. Create a new Web Services Feature Pack Deployment Manager profile:
Create > Feature Pack for Web Services > Deployment manager with Feature Pack for Web Services
3. Start the new deployment manager by changing to `was_install_dir/profiles/dmgr_profile_name/bin` and running `startManager.bin`

Federate your application server node into the deployment manager

The next step is to take the application server profiles that were created in the first stage after installing WAS, and to federate them into the deployment manager in order to provide managed nodes where you can create your cluster members:

1. Change to the directory for your standalone application server profile (`was_install_dir/profiles/appserver_profile_name/bin`)
2. Run the **addnode** command (`addnode hostname port`) to federate your node into the deployment manager, where *hostname* is the name of the host machine of your deployment manager and *port* is the port; for example, `addnode localhost 8879`
3. Log in to WAS administrative console for the deployment manager to check that it is federated (typically on `http://hostname:9060/admin`).
4. Go to **Servers > Application servers**, and delete the server that has just been federated (which should leave you with a Feature Pack for Web Services-enabled node).

Create the WAS Cluster

Now create the WAS cluster and servers into which you will deploy the WebSphere Business Events Runtime:

1. Log in to administrative console for the deployment manager.
2. Click **Servers > Clusters > New**
3. Name the new cluster `WBERuntime`
4. For each of the federated nodes, add a new cluster member:
 1. Create a cluster member, and name it appropriately; for example, `wbeserver1`, `wbeserver2`, `wbeserver3`.
 2. Select one of the managed nodes that you federated earlier (selecting a different node for each cluster member).

Setting up your repository

Set up your repository database as described in the WebSphere Business Events 6.2 information center (<http://publib.boulder.ibm.com/infocenter/wbevents/v6r2m0/topic/com.ibm.wbe.config.doc/doc/configuringtherepository.html>). The remainder of this document assumes that you are using a DB2 database for the repository.

Set up the WebSphere Application Server environment for WBE

The following steps take place in the WAS administrative console unless stated otherwise.

Set up WebSphere Business Events variables

This step creates the properties in WebSphere Application Server for the WebSphere Business Events Runtime to use.

Setting and verifying the WBE system properties

1. Navigate to the *install_dir/config/was* directory and edit the setenv script (setenv.bat on Windows, setenv.sh on Linux or UNIX). Information about each setting can be found within the setenv script. Taking a copy of the original setenv script will allow you to restore it if necessary. For further information about these settings, see the Configuring WebSphere Application Server information center topic.
2. Navigate to the *install_dir/config/db* folder and edit the setenv script. Edit the script to correspond to the repository database that you are using. More information about each setting can be found within the setenv script.
3. Navigate to the *install_dir/config/wbe* folder and edit the setenv script. Set the value of WBE_HOME to the home directory of the WebSphere Business Events installation.
4. On the first machine, where the deployment manager is running, create the default WebSphere Business Events properties:
 - a. At a command prompt, change directory to:

On Windows:

```
cd C:\Program Files\IBM\WBE62\config\wbe
```

On Linux and UNIX:

```
cd /opt/IBM/WBE62/config/wbe
```

- b. Run the update_wbeprops script that will call the update_wbeprops.py script:

On Windows:

```
update_wbeprops.bat
```

On Linux and UNIX:

```
./update_wbeprops.sh
```

5. In the WebSphere Application Server administrative console, check that the property values have been correctly set. This table uses an example where IBM DB2 stores your repository database (named WBEDB) and `db2inst1` is your user name:
 - a. Click **Resources > Resource Environment > Resource environment entries > WbeSrv01 > Custom properties**.
 - b. Set the property values as listed in the following table:

Property	Value
<code>as.director.common.db.default.dbclass</code>	<code>com.ibm.wbe.db.DB2DB</code>
<code>as.director.common.db.default.instance</code>	WBEDB
<code>as.director.common.db.default.dbport</code>	50000
<code>as.director.common.db.default.dbhost</code>	The host name of the machine that hosts the repository database.
<code>as.director.common.db.default.username</code>	<code>db2inst1</code>
<code>as.director.common.db.default.password</code>	Enter the password for the <i>db2inst1</i> account.
<code>as.director.server.jms.store.url</code>	<code>iiop://hostname:port</code> , where <i>hostname</i> and <i>port</i> are the connection details for the JMS server.
<code>log4j.appender.JMS1.ProviderURL</code>	<code>iiop://hostname:port</code> , where <i>hostname</i> and <i>port</i> are the connection details for the JMS server. For example, for SIBus, <code>iiop://localhost:2809</code> or for WebSphere MQ, <code>iiop://localhost:2810</code>
<code>log4j.appender.JMS2.ProviderURL</code>	<code>iiop://hostname:port</code> , where <i>hostname</i> and <i>port</i> are the connection details for the JMS server. For example, for SIBus, <code>iiop://localhost:2809</code> or for WebSphere MQ, <code>iiop://localhost:2810</code>

- c. Click **Apply**.
- d. Click **Save**.

Creating Repository Variables

1. Create the WebSphere Application Server environment variables:
 - a. Click **Environment > WebSphere Variables**.
 - b. From the **Scope** list click **Cell=cell-name**, where *cell-name* is the name of the cell in which you are creating the cluster.
 - c. Create each of the following environment variables: For each variable, click **New**, enter the details, then click **Apply**.

Variable	Value
DB2UNIVERSAL_JDBC_DRIVER_PATH	Enter the location of the database Java binaries. For example, on Linux: /opt/ibm/db2/V9.5/java
DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH	Enter the location of the database Java binaries. For example, on Linux: /opt/ibm/db2/V9.5/java

Setting other Environmental Variables

1. Set the JVM heap sizes. For each application server in the cluster:
 - a. Click **Servers > Application Servers > *server-name* > Java and Process Management > Process Definition > Java Virtual Machine**, where *server-name* is an application server in the cluster.
 - b. Set **Initial Heap Size** to 512.
 - c. Set **Maximum Heap Size** to 1024.
 - d. Click **Apply**.
 - e. Click **Save**.
2. Create the JVM custom property. For each application server in the cluster:
 - a. Click **Servers > Application Servers > *server-name* > Java and Process Management > Process Definition > Java Virtual Machine > Custom Properties**, where *server-name* is an application server in the cluster.
 - b. Click **New**.
 - c. In the **Name** field, type `wbe.home`
 - d. As the value, enter the location of the director directory in your WebSphere Business Events installation. For example:
On Windows: C:\Program Files\IBM\WBE62\director
On Linux and UNIX: /opt/IBM/WBE62/director
 - e. Click **Apply**.
 - f. Click **Save**.
3. Enable the Startup Bean service for each application server in the cluster:
 - a. Click **Servers > Application Servers > *server-name* > Container Services > Startup beans service**, where *server-name* is an application server in the cluster.
 - b. Select the **Enable service at server startup** check box.
 - c. Click **Apply**.
 - d. Click **Save**.

Create and configure SIB Messaging

This step will create and configure the Service Integration Bus components required by the WebSphere Business Events Runtime.

Create a new bus

1. Log on to the WAS administrative console.
2. Click **Service Integration > Buses**
3. Create a new bus called WbeBus.

Add bus members

1. Click on the new bus, WbeBus, which on the under **Service Integration > Buses** page, then click **Bus Members**.
2. Add a new bus member.
3. Click **Cluster > WBERuntime**
4. Click **Next**.
Click **Filestore**, then click **Next**.
5. Click **Log directory path**, then enter the following value:
\${USER_INSTALL_ROOT}/filestores/com.ibm.ws.sib/WBERuntime/log
6. Click **Permanent store directory path**, then enter the following value:
\${USER_INSTALL_ROOT}/filestores/com.ibm.ws.sib/WBERuntime/store
7. Click **Next**, then click **Finish**. Then click **Save**.

Add a new topic space

1. Click the **WbeBus**, then click **Destinations**.
2. Create a new destination
 1. Click **Topic Space**
 2. Enter WbeTopicSpace as the identifier.

Configure the JMS messaging provider

This step creates and configures the JMS resources that are needed by WebSphere Business Events Runtime.

Queue Connection Factories

1. In the administrative console, click **Resources > JMS > Queue connection factories**.
2. From the **Scope** menu, click **Cell**.
3. Create a new queue connection factory.
4. Set the following properties (leave everything else as the default value):
 1. Name: WbeQueueConnectionFactory
 2. JNDI Name: jms/WbeQueueConnectionFactory
 3. Bus: WbeBus
 4. Under **Provider endpoints**, list any hosts for servers within the cluster and their SIB_ENDPOINT_ADDRESS ports; for example, machine01:7276 Do not use localhost

Topic connection factories

1. In the administrative console, click **Resources > JMS > Topic connection factories**.
2. From the **Scope** list, click **Cell**.
3. Create a new topic connection factory.
4. Set the following properties (leave everything else as the default value):
 1. Name: WbeTopicConnectionFactory
 2. JNDI Name: jms/WbeTopicConnectionFactory
 3. Bus: WbeBus
 4. Under **Provider endpoints**, list any hosts for servers within the cluster and their SIB_ENDPOINT_ADDRESS ports; for example, machine01:7276 Do not use localhost.
 5. Under **Durable Subscription**, set the following properties:
 1. Client identifier: WBE
 2. Durable subscription home: Select the ME (for example, WBERuntime.000-WbeBus)
 6. Click **Apply**.
 7. Click **Connection pool properties**.
 1. Set **Maximum connections** to 30
 2. Set **Aged timeout** to 600

Topics

1. Click **Resources > JMS > Topics**
2. Delete any existing topics, then create the following topics at the scope of Cell. For each topic you create, select the default messaging provider:
 1. actionTopic
 1. Name: actionTopic
 2. JNDI Name: jms/actionTopic
 3. Topic name: actionTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace
 2. commandTopic
 1. Name: commandTopic
 2. JNDI Name: jms/commandTopic
 3. Topic name: commandTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace
 3. durableActionTopic
 1. Name: durableActionTopic
 2. JNDI Name: jms/durableActionTopic
 3. Topic name: durableActionTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace
 4. durableEventTopic
 1. Name: durableEventTopic
 2. JNDI Name: jms/durableEventTopic
 3. Topic name: durableEventTopic

4. Bus name: WbeBus
5. Topic space: WbeTopicSpace
5. eventLogTopic
 1. Name: eventLogTopic
 2. JNDI Name: jms/eventLogTopic
 3. Topic name: eventLogTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace
6. eventTopic
 1. Name: eventTopic
 2. JNDI Name: jms/eventTopic
 3. Topic name: eventTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace
7. historyModuleTopic
 1. Name: historyModuleTopic
 2. JNDI Name: jms/historyModuleTopic
 3. Topic name: historyModuleTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace
8. logTopic
 1. Name: logTopic
 2. JNDI Name: jms/logTopic
 3. Topic name: logTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace
9. replyTopic
 1. Name: replyTopic
 2. JNDI Name: jms/replyTopic
 3. Topic name: replyTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace
10. statusTopic
 1. Name: statusTopic
 2. JNDI Name: jms/statusTopic
 3. Topic name: statusTopic
 4. Bus name: WbeBus
 5. Topic space: WbeTopicSpace

Activation specifications

1. In the administrative console, click **Resources > JMS > Activation specifications**.
2. From the **Scope** list, click **Cell**.
3. Create the following new activation specifications, setting the following properties (leave everything else as the default value):
 1. wbe_events Activation Specification
 1. Name: wbe_events
 2. JNDI Name: jca/wbe_events
 3. Destination Type: Topic

4. Bus: WbeBus
5. Destination JNDI: jms/eventTopic
6. Under Additional:
 1. Max batch size = 1
 2. Max concurrent endpoints = 10
2. wbe_events_durable Activation Specification
 1. Name: wbe_events_durable
 2. JNDI Name: jca/wbe_events_durable
 3. Destination Type: Topic
 4. Bus: WbeBus
 5. Destination JNDI: jms/durableEventTopic
 6. Durability:
 1. Subscription Durability: Durable
 2. Subscription Name: WBE
 3. Client Identifier: WBE
 4. Durable Subscription Home: Select the ME e.g. WBERuntime.000-WbeBus
 7. Under Additional:
 1. Max batch size = 1
 2. Max concurrent endpoints = 10
3. wbe_history Activation Specification
 1. Name: wbe_history
 2. JNDI Name: jca/wbe_history
 3. Destination Type: Topic
 4. Bus: WbeBus
 5. Destination JNDI: jms/historyModuleTopic
 6. Under Additional:
 1. Max batch size = 1
 2. Max concurrent endpoints = 10
4. wbe_reset_watch Activation Specification
 1. Name: wbe_reset_watch
 2. JNDI Name: jca/wbe_reset_watch
 3. Destination Type: Topic
 4. Bus: WbeBus
 5. Destination JNDI: jms/actionTopic
 6. Under Additional:
 1. Max batch size = 1
 2. Max concurrent endpoints = 10

Repository

This step sets up the necessary details for the WBE repository. This step assumes that you are using DB2 as your repository database; for other databases, modify these steps as required.

Create JDBC Provider

1. In the administrative console, click **Resources > JDBC > JDBC Providers**.

2. From the **Scope** list, click **Cell** to create the JDBC provider at Cell scope.
3. Create a new JDBC provider.
4. Enter the following values:
 - a. Name: DB2 Universal JDBC Driver Provider
 - b. Database Type: DB2
 - c. Provider Type: DB2 Universal DB2 Provider
 - d. Implementation Type: Connection pool data source
5. Click **Next**, then enter the following values:
 - a. For the class path, enter the value:
 DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc.jar;\$
 {DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cu.jar;\$
 {DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cisuz.jar
 - b. Enter the location of the db2jcc.jar file and native library (this will be dependent on where the DB2 client was installed; for example, on Linux it might be /opt/ibm/db2/V9.5/java)
6. Save your settings.

Create the authentication data

This step is necessary to allow WAS to remotely log in to the repository database:

1. In the administrative console, click **Security > Secure administration, applications, and infrastructure > Java Authentication and Authorization Service > J2C authentication data**.
2. Click **New**.
3. In the **Alias** field, type the value WBEDB
4. In the **User ID** and **Password** fields, enter the user name and password that is required to access the repository database.
5. Enter a description of WBE database, and save your changes.

Create a JDBC data source

1. In the administrative console, click **Resources > JDBC > Data Sources**
2. From the **Scope** list, click **Cell** to create the data source at Cell scope.
3. Create a new data source.
4. Enter the following values:
 - a. Data source name: WBE Datasource
 - b. JNDI Name: jdbc/repository
 - c. Component-managed authentication alias and XA recovery authentication alias: WBEDB
5. Click **Next**, then enter the following values:
 - a. Database name:WBEDB
 - b. Driver Type: 4
 - c. For the Server name and port number fields, enter the appropriate values for the machine that hosts the repository database

Create the shared libraries

1. In the administrative console, click **Environment > Shared Libraries**
2. Select **Cluster=WBERuntime** as the scope to create the sheared libraries at Cluster scope.
3. Create the following new shared libraries:
 1. JDBC Driver:
 - a. Name: WbeJdbcDriver
 - b. Classpath: Enter location of the db2jcc.jar file e.g. /opt/ibm/db2/V9.5/java/db2cc.jar. The DB2 jar file should be in the same location on every machine in the cluster
 2. JWL Library:
 - a. Name: JWLLib
 - b. Classpath: \$
{WAS_INSTALL_ROOT}/optionalLibraries/IBM/JWL/2.0/odc-jsf.jar \$
{WAS_INSTALL_ROOT}/optionalLibraries/IBM/JWL/2.0/jsf-ibm.jar
4. Save your changes.

Installing the WBE runtime

This step installs the WBE runtime application into the clustered environment:

1. In the administrative console, click **Application > Install New Application**.
2. Click **Remote file system**.
3. Click **Browse** then click a node to browse to its file system.
4. Select the wberuntimeear application, which is in *wbe_install_dir/director/lib/wberuntimeear.ear*; for example, C:\Program Files\IBM\WBE62\director\lib\wberuntimeear.ear
5. In the **Context root** field, enter *wbe*
6. Click **Next**.
7. Select **Precompile JavaServer Pages Files (JSPs)**, then click **Next**.
8. In the **Clusters and Servers** field, select the **WBERuntime** cluster.
9. Select the check boxes for both the **WBERuntime** and **WebSphere Business Events** modules, and click **Apply**.
10. Click **Finish** and save your changes to complete the runtime installation.

Ensuring that changes have been saved

To ensure that all changes are propagated, before starting all the application servers:

1. Stop all the application servers.
2. Start the deployment manager.

This will ensure that all configuration changes are applied.