Blueprint 6.0

Installation Guide

3/31/2015

blueprint

Contents

Overview

This *Blueprint Installation Guide* walks you through the Blueprint installation process. Please refer to the *Blueprint Upgrade Guide* if Blueprint is already installed and you want to upgrade to a newer version.

Before you begin installing Blueprint:

- Ensure that you have your Blueprint activation key. If you don't have an activation key, please submit a case to Blueprint Support using the customer portal at http://portal.blueprintsys.com/
- Ensure that your environment meets the minimum system requirements.

System requirements

Web application server and database server requirements

The web application server must have access to communicate with the database server. Refer to the User requirements section for more information.

The following table outlines the hardware and software requirements for both the distributed-server and single-server configurations.

	Web application server (distributed-server configuration)	Database server (distributed-server configuration)	Web application server and database server (single-server configuration)
Hardware requirements	■ 2.0 GHz or faster ■ Multi-core CPU ■ 8 GB RAM minimum (12 GB recommended)	■ 2.0 GHz or faster ■ Multi-core CPU ■ 8 GB RAM minimum (16 GB recommended) ■ 50 GB HDD space minimum (100 GB recommended for database and backups)	■ 2.0 GHz or faster ■ Multi-core CPU ■ 12 GB RAM minimum (24 GB recommended) ■ 50 GB HDD space minimum (100 GB recommended for database and backups)

	Web application server (distributed-server configuration)	Database server (distributed-server configuration)	Web application server and database server (single-server configuration)

Software requirements	▪ One of:
	  ▪ Windows Server 2012 with IIS 8
	  ▪ Windows Server 2008R2 with IIS 7.5
	  ▪ Windows Server 2008 with IIS 7
	▪ .NET Framework 4.5
		▪ One of:
		  ▪ Windows Server 2012 R2 (standard edition or enterprise)
		  ▪ Windows Server 2008 R2 (standard edition or enterprise)
		  ▪ Windows Server 2008
		  ▪ Windows Server 2003
		▪ One of:
		  ▪ SQL Server 2014
		  ▪ SQL Server 2012
		  ▪ SQL Server 2008R2 (standard edition and above)
		  ▪ SQL Server 2008
			▪ One of:
			  ▪ Windows Server 2012 with IIS 8.5
			  ▪ Windows Server 2012 with IIS 8
			  ▪ Windows Server 2008 R2 with IIS 7.5
			  ▪ Windows Server 2008 with IIS 7
			▪ One of:
			  ▪ SQL Server 2014
			  ▪ SQL Server 2012
			  ▪ SQL Server 2008 R2 (standard edition and above)
			  ▪ SQL Server 2008
			▪ .NET Framework 4.5

Note: If you are also installing services, such as job services or HP Quality Center legacy support, we recommend having an additional 4 GB RAM for operations on the applicable server.

Requirements for ALM integrations via OpsHub

There are additional requirements for environments licensed to integrate bi-directionally with ALM systems. This functionality is separate from native integrations and is facilitated by OpsHub.

The following requirements should be met for environments licensed to integrate via OpsHub:

	ALM integrations server
Hardware requirements	■ 2.0 GHz or faster ■ Multi-core CPU ■ 2 GB RAM minimum ■ 15 GB HDD space minimum

Note: If you are hosting the ALM integrations server on the same server as the web application or database, simply add the specifications above to the applicable server specifications outlined in Web application server and database server requirements.

Web server configuration requirements

IIS server configuration requirements

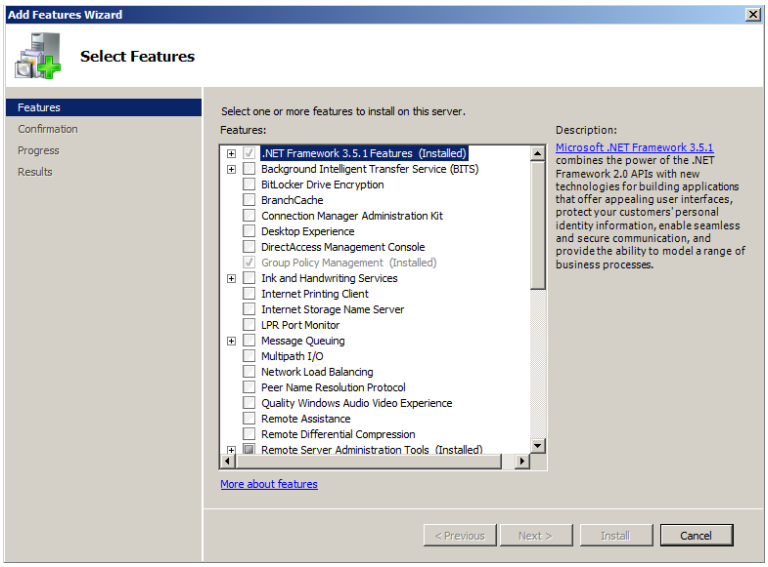
To see your IIS configuration requirements, select the version of Windows Server you are using:

- Windows Server 2008
- Windows Server 2012

Windows Server 2008

There are a number of IIS Features and Role services that must be enabled to install Blueprint.

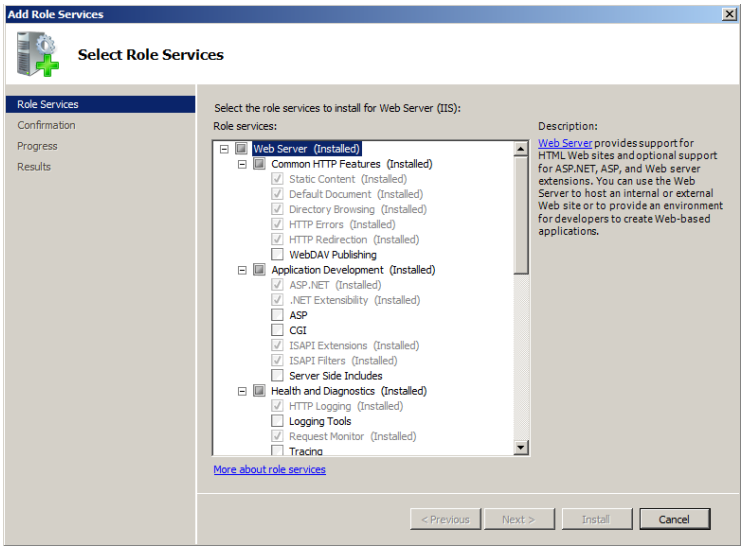
Features



The following Features must be enabled for Windows Server 2008:

- .NET Framework 3.5.1 Features
  - .NET Framework 3.5.1
  - WCF Activation
    - HTTP Activation
    - Non-HTTP Activation
- Remote Server Administration Tools
  - Role Administration Tools
    - Web Server (IIS) Tools
- Windows Process Activation Service
  - Process Model
  - .NET Environment
  - Configuration APIs

Role Services



The following Role Services must be enabled for Windows Server 2008:

- Common HTTP Features
  - Static Content
  - Default Content
  - Directory Browsing
  - HTTP Errors
  - HTTP Redirection
- Application Development
  - ASP.NET
  - .NET Extensibility
  - ISAPI Extensions
  - ISAPI Filters
- Health and Diagnostics
  - HTTP Logging
  - Request Monitor

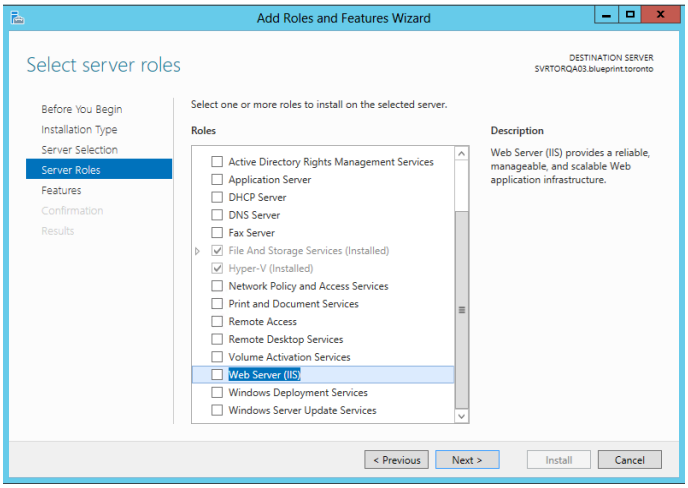
- Security
  - Basic Authentication
  - Windows Authentication
  - Digest Authentication
  - Request Filtering
  - IP and Domain Restrictions
- Performance
  - Static Content Compression
  - Dynamic Content Compression
- Management Tools
  - IIS Management Console
  - IIS Management Scripts and Tools

Note: World Wide Web Publishing Service must also be started within *Services* (Control Panel > Administrative Tools).

Windows Server 2012

There are a number of Roles, Features and Role Services that must be enabled in the IIS *Add Roles and Features Wizard* before you can use Blueprint.

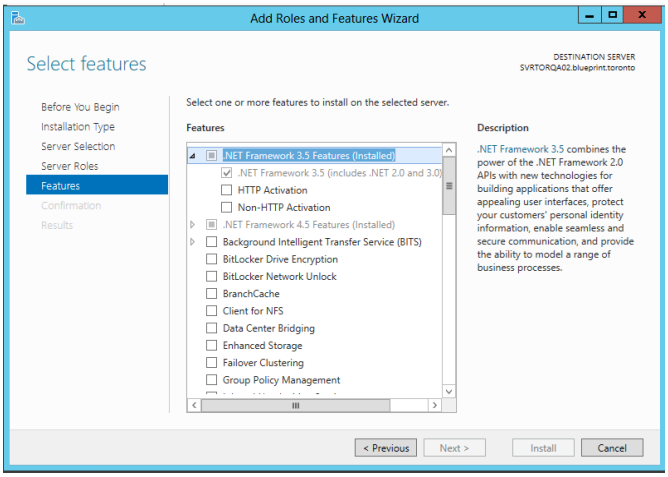
Server roles



Within *Server Manager > Manage > Add Roles and Features*, the following Roles must be enabled for Windows Server 2012:

- Web Server (IIS)

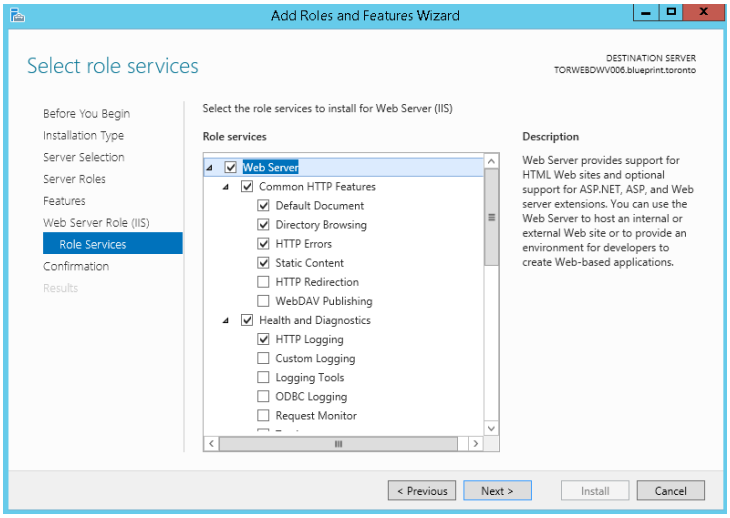
Features



The following Features must be enabled for Windows Server 2012:

- .NET Framework 3.5 Features
  - .NET Framework 3.5
  - HTTP Activation
  - Non-HTTP Activation
- .NET Framework 4.5 Features
  - .NET Framework 4.5
  - ASP.NET 4.5
  - WCF Services
  - HTTP Activation
- Windows Process Activation Service
  - Process Model
  - .NET Environment 3.5
  - Configuration APIs

Role services



The following Role Services must be enabled for Windows Server 2012:

- Web Server
  - Common HTTP Features
    - Default Document
    - Directory Browsing
    - HTTP Errors
    - Static Content
    - HTTP Redirection
  - Health and Diagnostics
    - HTTP Logging
    - Request Monitor
  - Performance
    - Static Content Compression
    - Dynamic Content Compression
  - Security
    - Request Filtering
    - Basic Authentication
    - Digest Authentication
    - IP and Domain Restrictions
    - Windows Authentication
  - Application Development
    - .NET Extensibility 3.5
    - .NET Extensibility 4.5
    - ASP
    - ASP .NET 3.5
    - ASP .NET 4.5
    - ISAPI Extensions
    - ISAPI Filters

- Management Tools
  - IIS Management Console
  - IIS Management Scripts and Tools

Note: World Wide Web Publishing Service must also be started within *Services* (Control Panel > Administrative Tools).

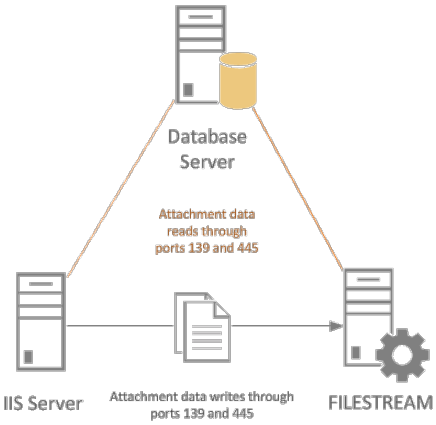
Database server configuration requirements

To complete database server configuration, perform the following steps:

- Open ports 139, 445 and 1433 in Firewall
- Enabling FILESTREAM
- Verifying FILESTREAM is enabled

FILESTREAM

FILESTREAM must be enabled in order to use Blueprint. The following diagram shows how the IIS server and the database server communicate with FILESTREAM:



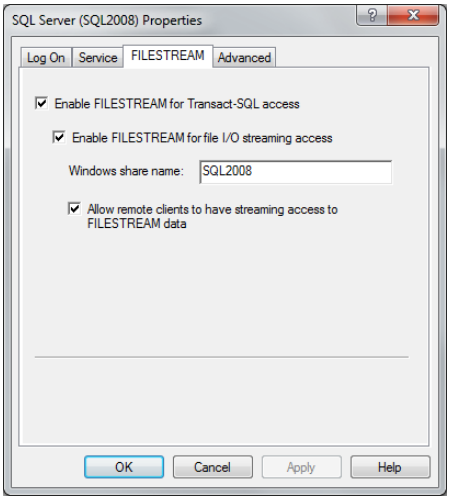
Enabling FILESTREAM

To enable FILESTREAM, perform the following steps:

Important

Ports 139 and 445 must be open between the Blueprint Application Server and the Database Server.

1. In *SQL Server Configuration Management*, select **SQL Server Services**.
2. Select your SQL server and select **Properties** from context menu.
3. Select the *FILESTREAM* tab:



4. Enable the following options:
   - **Enable FILESTREAM for Transact-SQL access**
   - **Enable FILESTREAM for file I/O streaming access**
   - **Allow remote clients to have streaming access to FILESTREAM data**

     Note: This option is only required if you install Blueprint in a distributed manner (separate Web Application server and Database server).
5. Click **OK**.

Next, verify that FILESTREAM is enabled.

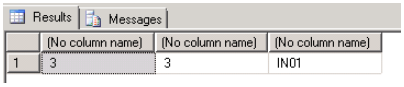
Verifying FILESTREAM is enabled

To verify FILESTREAM is enabled, perform the following steps:

1. Open SQL Server Management Studio.
2. To verify whether FILESTREAM has been enabled, run the following `SELECT` statement:

```
SELECT
SERVERPROPERTY ('FilestreamConfiguredLevel'),
SERVERPROPERTY ('FilestreamEffectiveLevel'),
SERVERPROPERTY ('FilestreamShareName');
```

   If FILESTREAM has been successfully enabled, the following output appears:



   - If the first or second number is “0”, FILESTREAM is not enabled on the server and the following SQL statement must be run:

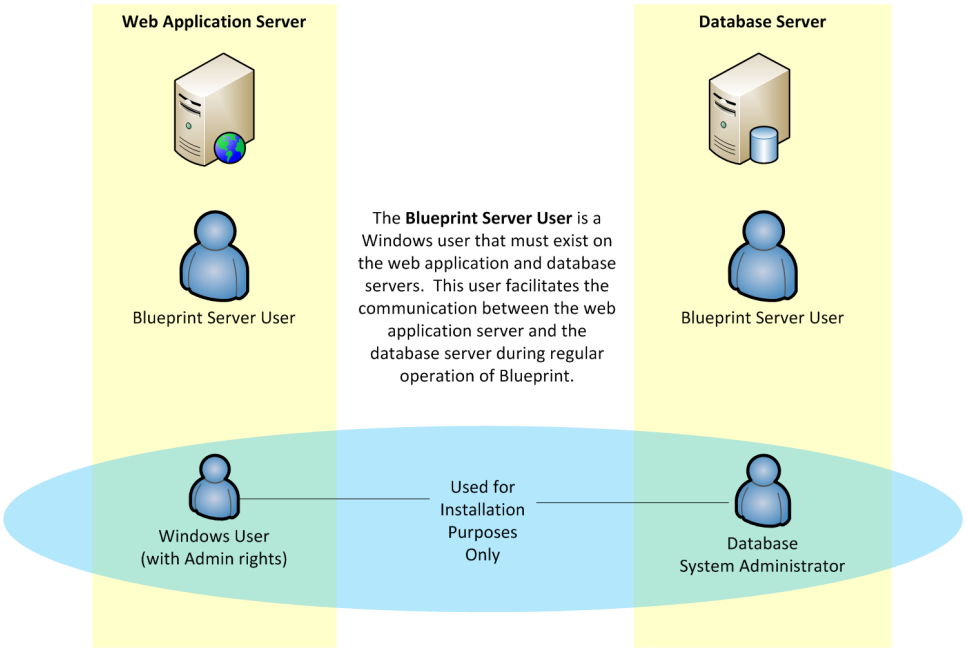
```
EXEC sp_configure filestream_access_level, 3
```

```
RECONFIGURE
```

To verify FILESTREAM has been successfully configured, re-run the SELECT statement.

User requirements

The following diagram illustrates the Blueprint installation user requirements:



Blueprint Server User

The *Blueprint Server User* is a service account for the application pool connection from IIS to the database.

Note: The Blueprint Server User is required for everyday Blueprint operation.

The following requirements must be met:

- This user account must be defined as a corporate *domain* user (that is, domain_name\username) or a local server account (that is, ApplicationPoolIdentity).
- This user account must exist on *both* the web application server and the database server.
- This user account can have standard user privileges and does not need to be an administrator. To enable Blueprint server logging, the Blueprint Server User must have **`modify+read+write`** permissions on the Blueprint program data folder (example: **`C:\ProgramData\Blueprint Software Systems`**). You can grant these permissions by including the Blueprint Server User in a group that already has access to the folder, or by granting the permissions directly to the user.

Windows User

The *Windows User* is a standard windows login which is used to install the Blueprint application. This user should have local Administrator privileges to the web application server (that is, IIS server) where Blueprint is to be installed. The *Windows User* account is required for installation purposes only, and is not used under regular operation of Blueprint.

Database System Administrator

The *Database System Administrator* is an account defined within the SQL server database. This account must have administrator privileges in order to create and update tables within the database server. This account is typically a domain account. However, if the database server is not part of a domain, then the default database administrator account is "sa" .
During installation, you must provide the credentials of this *Database System Administrator* account so the Blueprint installer can create the database and grant appropriate permissions to the *Blueprint Server User*. The *Database System Administrator* account is required for installation purposes only, and is not used under regular operation of Blueprint.

Client requirements

The following requirements must be met on each client system:

	Client System
Hardware requirements	■ 1.6 GHz or faster Pentium 4-class CPU (multi-core recommended) ■ 2 GB RAM minimum (3 GB recommended)

	Client System

Software requirements	■ Supported operating systems:
	  ■ Windows 8
	  ■ Windows 7
	  ■ Windows Vista
	  ■ Windows XP
	  ■ Mac OS 10.9 and later
	■ Supported web browsers:
	  ■ Microsoft Internet Explorer version 8 and later (compatibility mode is not supported)
	  ■ Mozilla Firefox version 19 and later
	  ■ Google Chrome version 25 and later
	  ■ Safari 7 and later
	■ Silverlight 5
	■ Minimum screen resolution of 1280x1024 (1600x900 or higher recommended)
	The following system requirements only apply if you plan to use the Blueprint template authoring add-in for template development purposes:
	■ .NET Framework 4
	■ Microsoft Office 2007, 2010 or 2013

Installation Steps

Step 1: Install Blueprint on the web application server

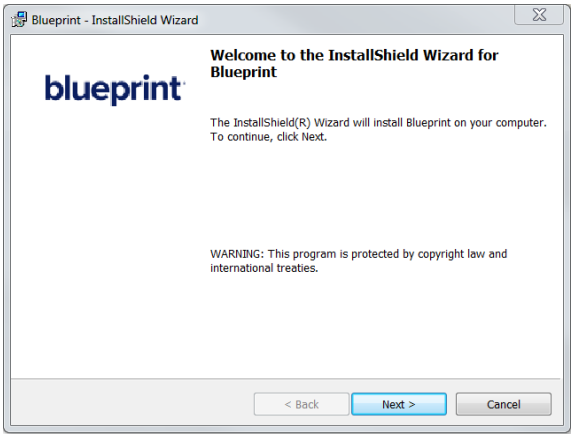
- Continue to the Automatic Installation section (recommended) if you want to install Blueprint using the Blueprint InstallShield Wizard. After the installation is completed, the Configuration Wizard automatically opens (Step 2).
- Continue to the Blueprint Files Setup section if you want to deploy the Blueprint files and then manually configure the application and database using the command line utilities.

After Blueprint is installed, continue to Step 2: Configure the web application, database and services.

Option 1: Automatic Installation

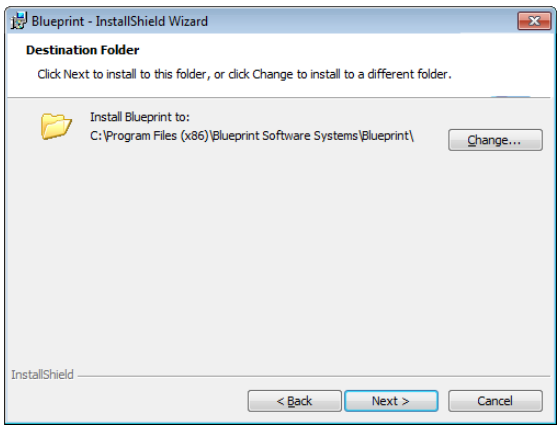
`BlueprintSetup.exe` extracts all of the application files and configuration utilities that are required for the Blueprint installation.

After you launch the **`BlueprintSetup.exe`** file, the following InstallShield Wizard appears:



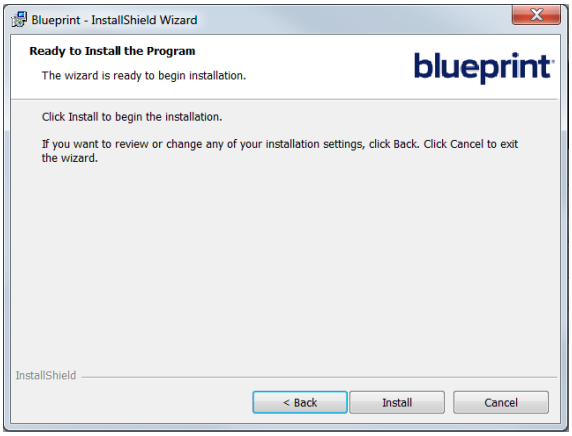
Click **Next** to continue.

On the next dialog, you can configure the Blueprint installation directory.



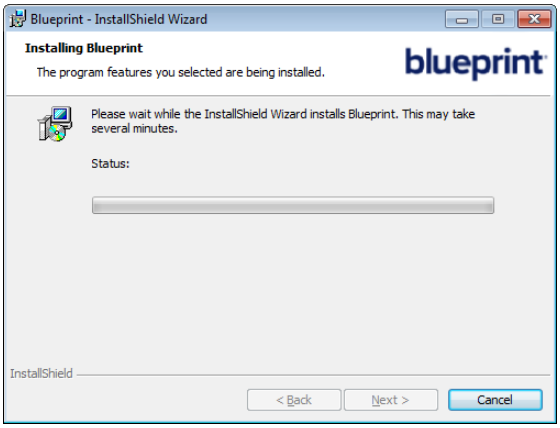
Click **Next** to continue.

The following wizard appears to start the installation process:

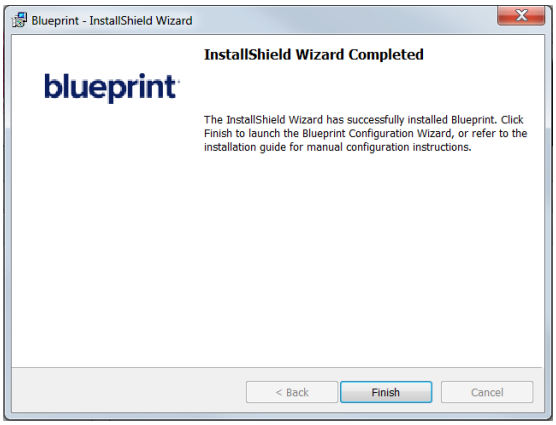


Click **Install** to proceed with the Blueprint installation.

The following dialog is displayed while the installation is in progress.



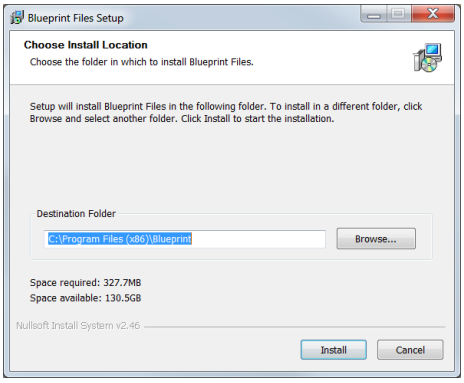
When the installation is complete, the following dialog is displayed:



Option 2: Blueprint Files Setup (Manual Installation)

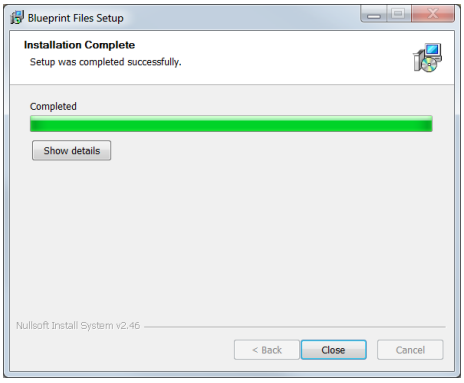
Run **`BlueprintFiles.exe`**.

`BlueprintFiles.exe` extracts all of the application files and configuration utilities that are required for the Blueprint installation. After you open **`BlueprintFiles.exe`**, the following Files Setup Wizard appears:



Click **Install** to continue.

When the installation has completed, the following dialog appears:



Click **Close**.

You have successfully deployed the setup files to the web application server.

Step 2: Configure the web application, database and services

- Continue to the Blueprint Configuration Wizard section (recommended) to configure Blueprint using the Configuration Wizard.
- Continue to the Advanced / Manual Configuration section to configure the application, database and services using the command line configuration utilities.

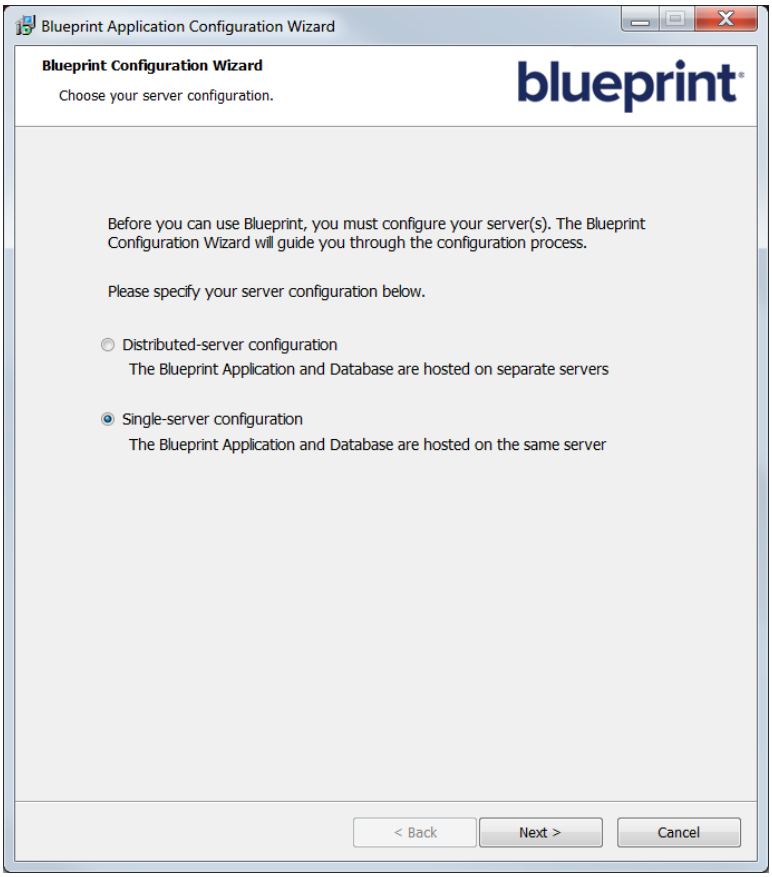
After configuration is completed, continue to Step 3: Activate Blueprint.

Option 1: Blueprint Configuration Wizard

The Blueprint Configuration Wizard launches automatically after installing Blueprint using the installation wizard.

> **Note**: The Blueprint Configuration Wizard is only intended for the initial configuration of Blueprint. Refer to the Appendix to learn more about operations that you may need to run after the initial configuration.

The Blueprint Configuration Wizard dialog allows you to choose your desired configuration:



Select one of the following options and then click **Next**:

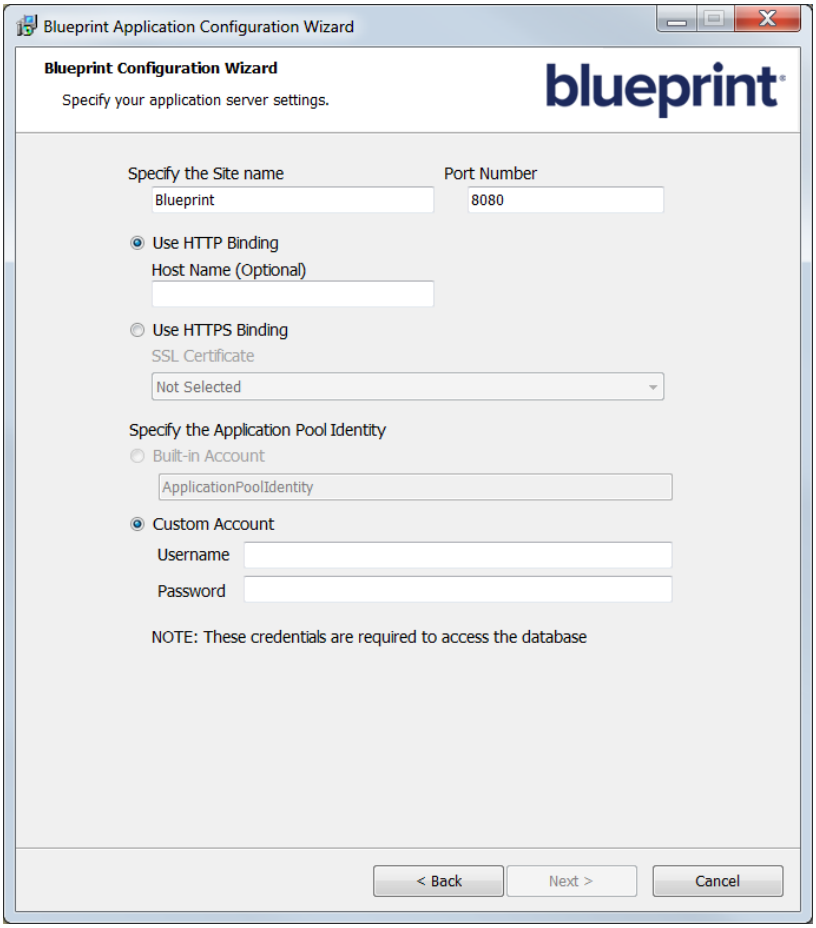
- **Distributed-server configuration**: This option allows you to host the Blueprint application and database on separate servers. This is the typical configuration. If you choose this option, refer to the distributed-server configuration section below.

- **Single-server configuration**: This option allows you to host the Blueprint application and database on the same server. This configuration can be useful for demo purposes. If you choose this option, refer to the single-server configuration section below.

Distributed-server configuration

Application Configuration (Distributed)

The Application (IIS AppPool and Site) configuration dialog looks like this:



Complete the following fields and then click **Next**:

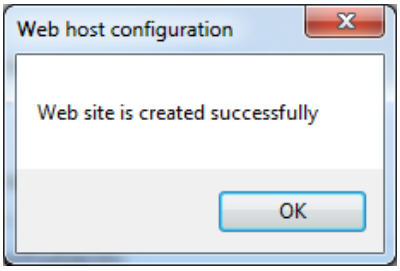
- **Specify the Site name**: Define the name of the IIS Site. Example: Blueprint
- **Port Number**: Define the port number of the IIS Site. Example: 8080
- Specify the Binding Type:
  - **Use HTTP Binding**: Select this option if you want to run Blueprint over HTTP.
  - **Use HTTPS Binding**: Select this option if you want to run Blueprint over HTTPS. If you choose HTTPS Binding, you must specify an **SSL Certificate**.
- Specify the Application Pool Identity:

Note: The Custom Account refers to the user account that was outlined in the User Prerequisites section of this document. Example: acme\rrunner.

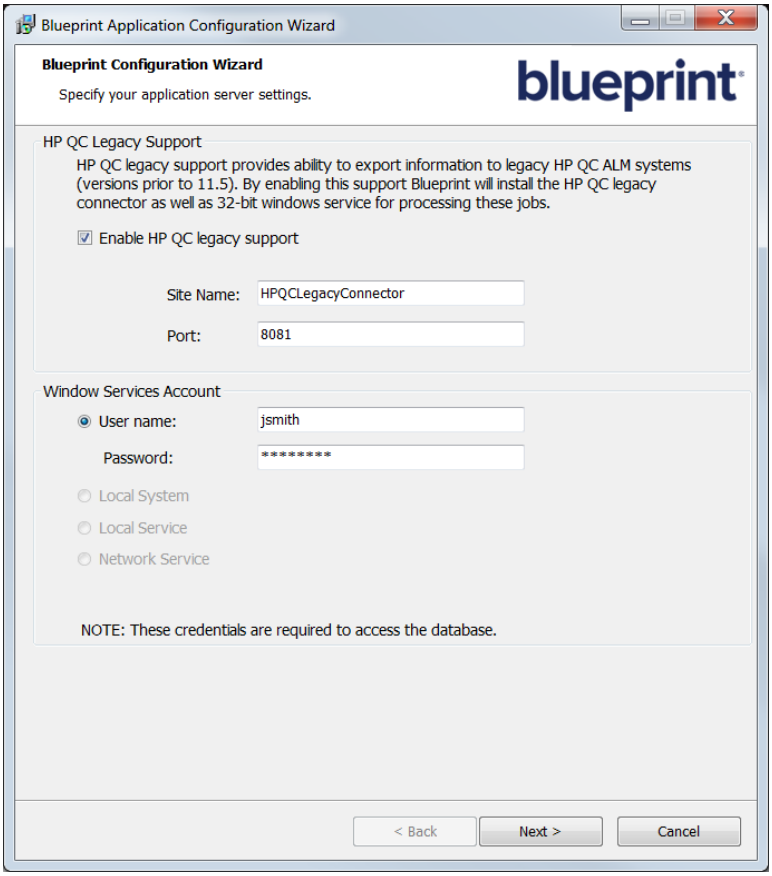
- **Username**: Specify the name of the Blueprint Server User account that has appropriate permissions on both the application and database servers.
- **Password**: Specify the password for the Blueprint Server User account.

Important: Please ensure that you type the correct password. The password is not verified by the wizard.

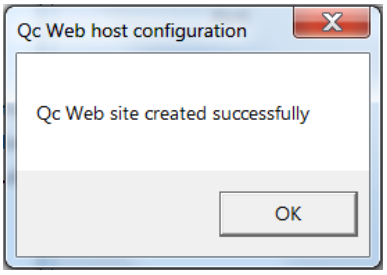
The following dialog is displayed after the site is created successfully:



The *Blueprint Application Configuration Wizard* dialog gives you the option of enabling legacy support for version 12 and earlier of HP Quality Center:

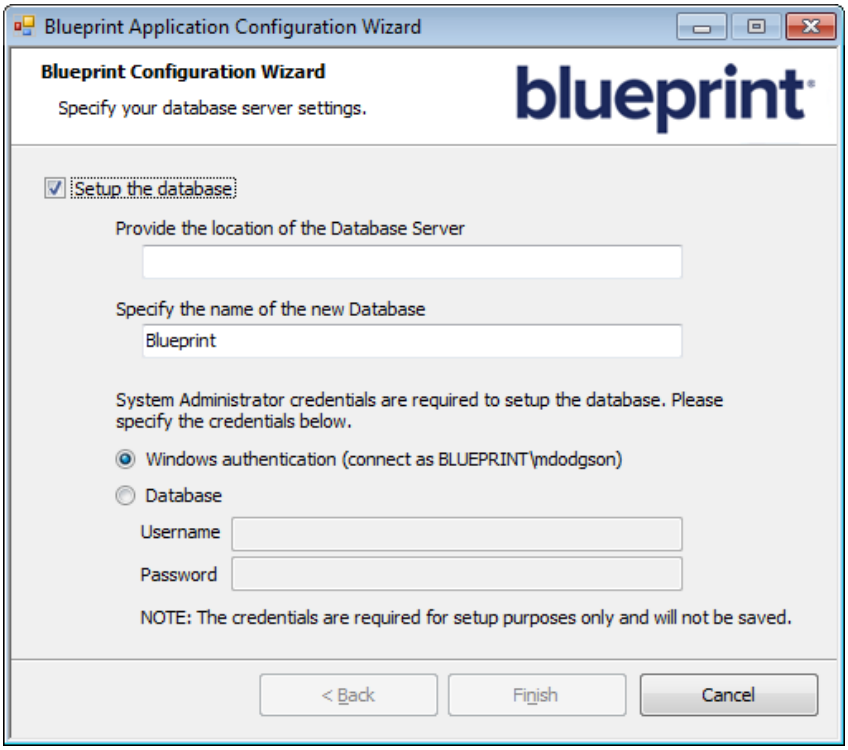


If you enabled legacy support for HP Quality Center and the legacy web site is successfully created, a dialog appears:

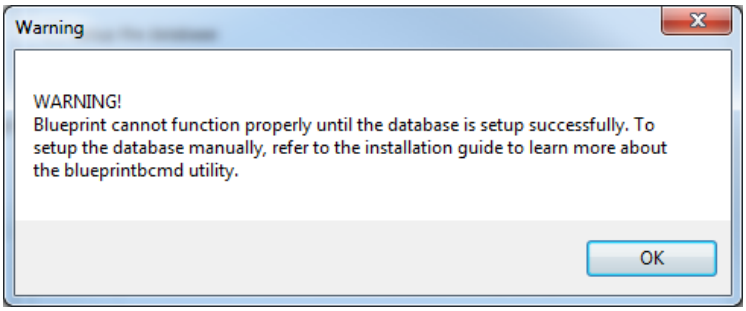


Database Configuration (Distributed)

The Database configuration dialog looks like this:



The database must be setup successfully before you can use Blueprint. If you clear the **Setup the database** option, the following dialog is displayed:



If you choose to setup the database using the Blueprint Configuration Wizard, complete the following fields and then click **Next**:

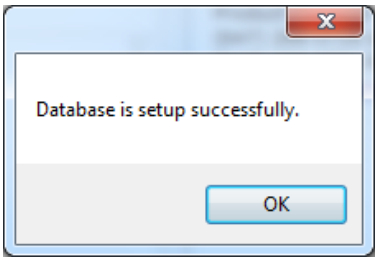
- **Provide the location of the Database Server**: The database server must be provided in the following format: [server_name]\[instance_name]
  Example: DBSERVER\INSTANCE01
- **Specify the name of the new Database**: The database must not already exist.

  Example: Blueprint
- System Administrator Credentials: Choose whether you wish to authenticate using *Windows authentication* or *Database* authentication.

  Important: This user must be a Database System Administrator (SA). This allows the installer to create the new database and grant permissions to the database so the web application server can access the database. The SA credentials are not stored anywhere in the system. The SA user account is only required for installation purposes and is not used during normal operation of the application. During normal operation of Blueprint, the Blueprint Server User account (example: acme\rrunner) is used to facilitate communication between the web application and database servers.

  - **Windows authentication**: If you choose windows authentication, the wizard automatically uses the user that is currently logged in. You must ensure that the user has SA privileges on the database server.
  - **Database authentication**: If you choose database authentication, you must specify the username and password of an account that has SA privileges on the database server.

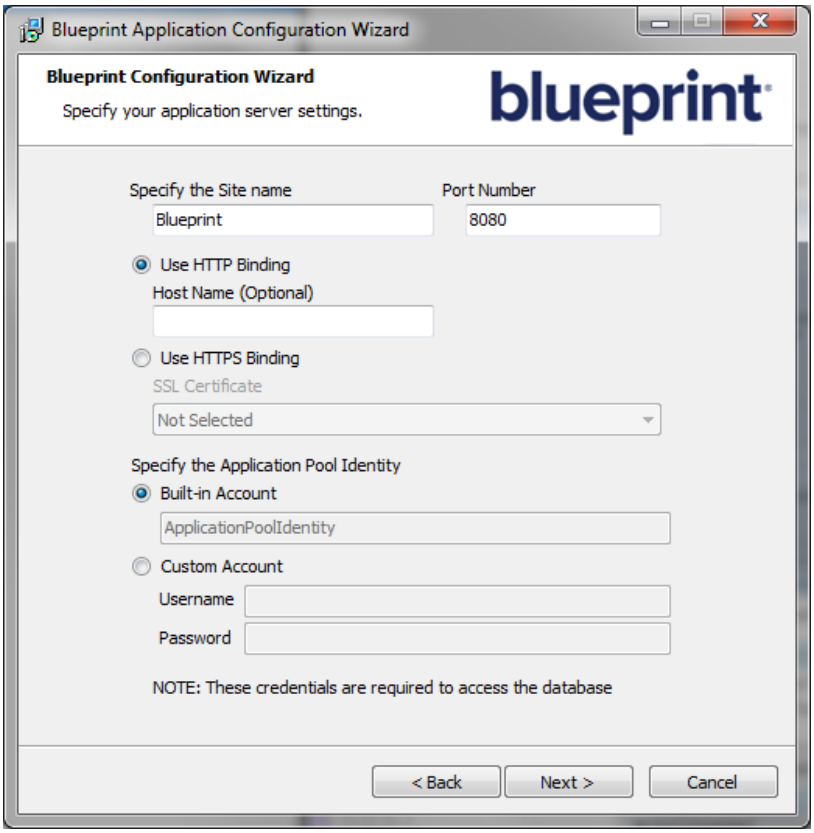
The following dialog is displayed after the database is setup successfully:



Single-server configuration

Application Configuration (Single-Server)

The Application (IIS AppPool and Site) configuration dialog looks like this:



Complete the following fields and then click **Next**:

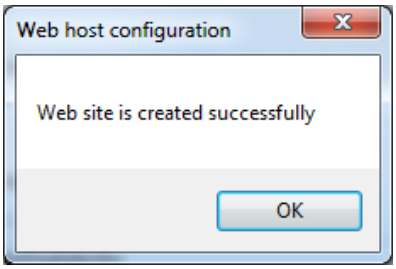
- **Specify the Site name**: Define the name of the IIS Site. Example: Blueprint
- **Port Number**: Define the port number of the IIS Site. Example: 8080
- Specify the Application Pool Identity:
  - **Built-in Account**
  - **Custom Account**

> **Note**: The Custom Account refers to the user account that was outlined in the User Prerequisites section of this document. Example: acme\rrunner.

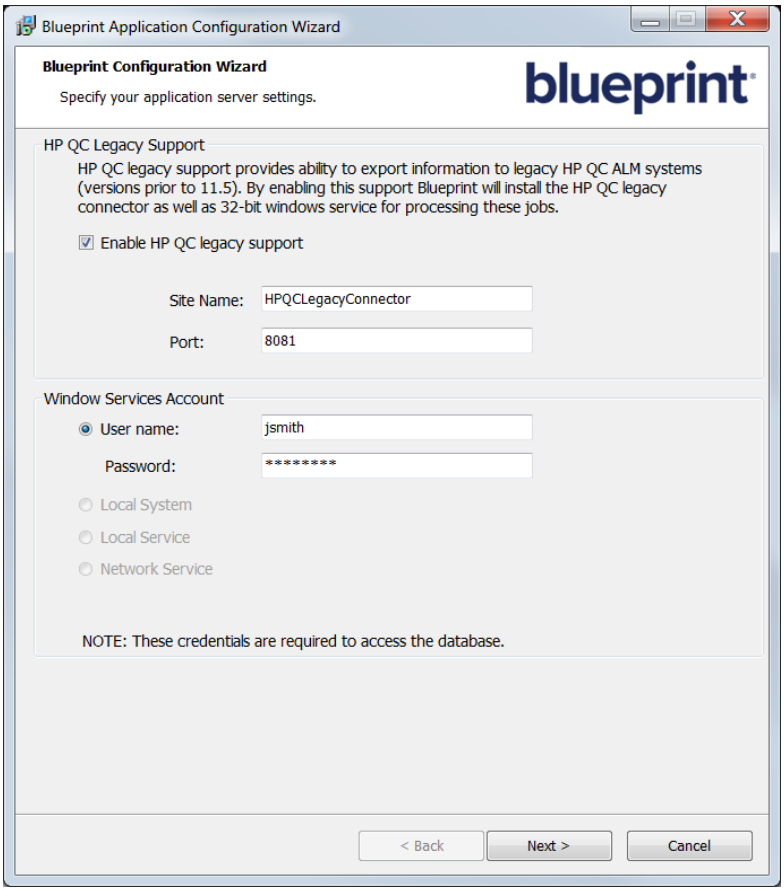
  - **Username**: Specify the name of the Blueprint Server User account that has appropriate permissions on both the application and database servers.
  - **Password**: Specify the password for the Blueprint Server User account.

> **Important**: Please ensure that you type the correct password. The password is not verified by the wizard.

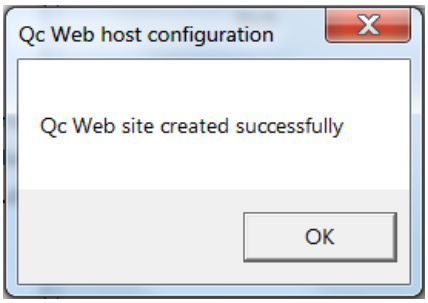
The following dialog is displayed after the site is created successfully:



The *Blueprint Application Configuration Wizard* dialog gives you the option of enabling legacy support for version 12 and earlier of HP Quality Center:

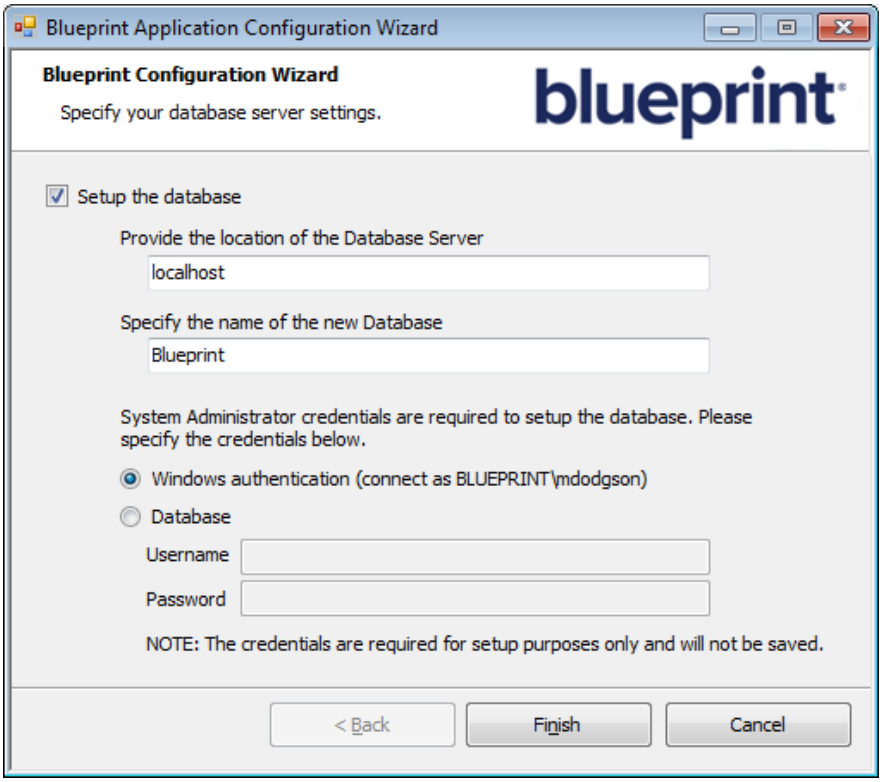


If you enabled legacy support for HP Quality Center and the legacy web site is successfully created, a dialog appears:

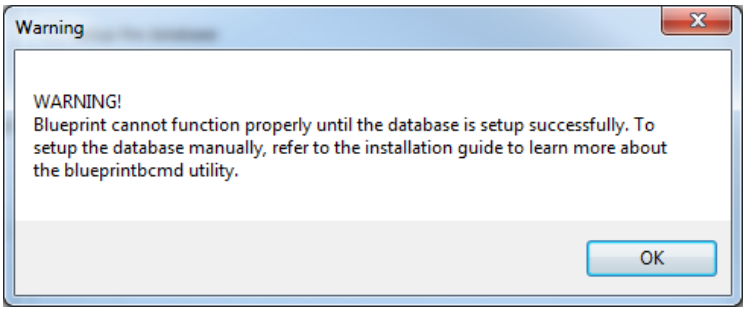


Database Configuration (Single-Server)

The *database configuration* dialog looks like this:



The database must be setup successfully before you can use Blueprint. If you clear the **Setup the database** option, the following dialog is displayed:



If you choose to setup the database using the Blueprint Configuration Wizard, complete the following fields and then click **Next**:

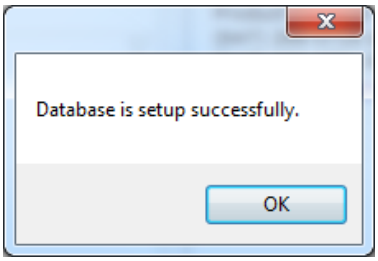
- **Provide the location of the Database Server**: The database server must be provided in the following format: [server_name]\[instance_name]
  Example: DBSERVER\INSTANCE01
- **Specify the name of the new Database**: The database must not already exist.

  Example: Blueprint
- System Administrator Credentials: Choose whether you wish to authenticate using *Windows authentication* or *Database* authentication.

Important: This user must be a Database System Administrator (SA). This allows the installer to create the new database and grant permissions to the database so the web application server can access the database. The SA credentials are not stored anywhere in the system. The SA user account is only required for installation purposes and is not used during normal operation of the application. During normal operation of Blueprint, the Blueprint Server User account (example: acme\rrunner) is used to facilitate communication between the web application and database servers.

- **Windows authentication**: If you choose windows authentication, the wizard automatically uses the user that is currently logged in. You must ensure that the user has SA privileges on the database server.
- **Database authentication**: If you choose database authentication, you must specify the username and password of an account that has SA privileges on the database server.

The following dialog is displayed after the database is setup successfully:



Option 2: Advanced / manual configuration

This section explains how to setup and configure the web application and database using the configuration utilities (instead of using the *Blueprint Configuration Wizard*). The configuration utilities must be run on the web application server with Administrator privileges.

Note: If you completed the full installation using the Wizard, you are NOT required to run the configuration utilities. The configuration utilities are run automatically when you complete the installation using the wizard. You are only required to run the configuration utilities manually if you decide to perform a manual install instead of using the Wizard.

The configuration utilities are located in a folder called **`Setup`**. The **`Setup`** folder is located in the directory that you chose for installing Blueprint. For example, the default path is:

```
C:\Program Files\Blueprint Software Systems\Blueprint\Setup
```

There are two configuration utility files:

- `blueprintwebcmd.exe`
- `blueprintdbcmd.exe`

Tip

You can type the following commands to view more information about the command parameters:

```
blueprintwebcmd.exe /help
```

```
blueprintdbcmd.exe /help
```

Configure the web application

Here's a summary of the steps that must be completed to configure the web application:

- Create the Application Pool:
  This step involves creating an Application Pool.
- Create the Site:
  This step involves creating a Site with the same name as the Application Pool.
- Configure the connection string:
  This step tells the application where the database is located and how to communicate with the database.

Below is an example of the commands that can be run to configure the web application:

> **Important**: The commands must be run with Administrator privileges on the web application server. Refer to the Configuration Utility Command Reference section for more information about these commands.

1. Create the Application Pool:

```
blueprintwebcmd.exe /object APPPOOL /command ADD /apppoolname
[Blueprint] /userid [acme\rrunner] /password [password]
```

2. Configure the Site:

```
blueprintwebcmd.exe /object SITE /command ADD /wsname [Blueprint]
/port [8080] /dir [C:\Program Files (x86)\Blueprint Software
Systems\Blueprint\Web] /apppoolname [Blueprint]
```

3. Configure the connection string:

```
blueprintwebcmd.exe /object DBCONFIG /command SET /datasource
[DBSERVER\INSTANCE01] /wsname [Blueprint] /integratedsec TRUE /catalog
[BlueprintDB]
```

Unlock the website security

```
%windir%\system32\inetsrv\appcmd unlock config %WSNAME%
/section:windowsAuthentication /commit:apphost
%windir%\system32\inetsrv\appcmd unlock config %WSNAME%
/section:basicAuthentication /commit:apphost
%windir%\system32\inetsrv\appcmd unlock config %WSNAME%
/section:digestAuthentication /commit:apphost
%windir%\system32\inetsrv\appcmd unlock config %WSNAME%
/section:anonymousAuthentication /commit:apphost
%windir%\system32\inetsrv\appcmd unlock config %WSNAME%
/section:system.webServer/serverRuntime /commit:apphost
```

...where %WSNAME% is the name of the Blueprint site

Configure the database

Here's a summary of the steps that must be completed to configure the database:

- Create the database:
  This step involves creating a database shell, which is a blank container and does not contain any content.
- Add security (that is, add user to the database):
  This step adds the Blueprint Server User to the database and grants appropriate permissions to the user.
- Initialize the database:
  This step creates the database schema and pre-populates the database with the data that is necessary to start using Blueprint.

Below is an example of the commands that can be run to configure the database:

Important: The commands must be run with Administrator privileges on the web application server.

Note: If you want to use *Windows Authentication* instead of database authentication, you can set the `integratedsec` parameter to *TRUE* in the commands below and omit the `userid` and `password` parameters. Refer to the Configuration Utility Command Reference for more information about these commands.

1. Create the database:

```
blueprintdbcmd.exe /object DB /command ADD /datasource
[DBSERVER\INSTANCE01] /catalog [BlueprintDB] /integratedsec FALSE
/userid [dbadmin] /password [pAssw0rd]
```

2. Add security (that is, add the Blueprint Server User to the database):

```
blueprintdbcmd.exe /object USER /command ADD /datasource
[DBSERVER\INSTANCE01] /catalog [BlueprintDB] /integratedsec FALSE
/nuseridentity [acme\rrunner] /userid [dbadmin] /password [pAssw0rd]
```

3. Initialize the database:

```
blueprintdbcmd.exe /object DB /command INIT /datasource
[DBSERVER\INSTANCE01] /catalog [BlueprintDB] /integratedsec FALSE
/userid [dbadmin] /password [pAssw0rd]
```

4. Stop and then restart the Blueprint application pool and Blueprint web site.

   Note: Your Blueprint application pool and Blueprint web site may have different names, depending on what you chose during installation.

Configure services

Configuring services is necessary to take advantage of the following functionality in Blueprint:

- Generating documents
- Exporting artifacts to ALM systems
- Generating tests
- HP Quality Center legacy support for versions 12 and earlier

To install services, see Installing services manually.

Note: After finishing the services installation, proceed to Step 3: Activate Blueprint.

Step 3: Activate Blueprint

Prior to activating Blueprint, you will see the following message if you navigate to the Blueprint URL (example: http://localhost:8080/):

`The application cannot be run without a valid license. Please contact your Administrator.`

You must activate Blueprint using one of the following methods:

- Online activation via internet (recommended)
- Manual offline activation via Email

Activating Blueprint using online activation

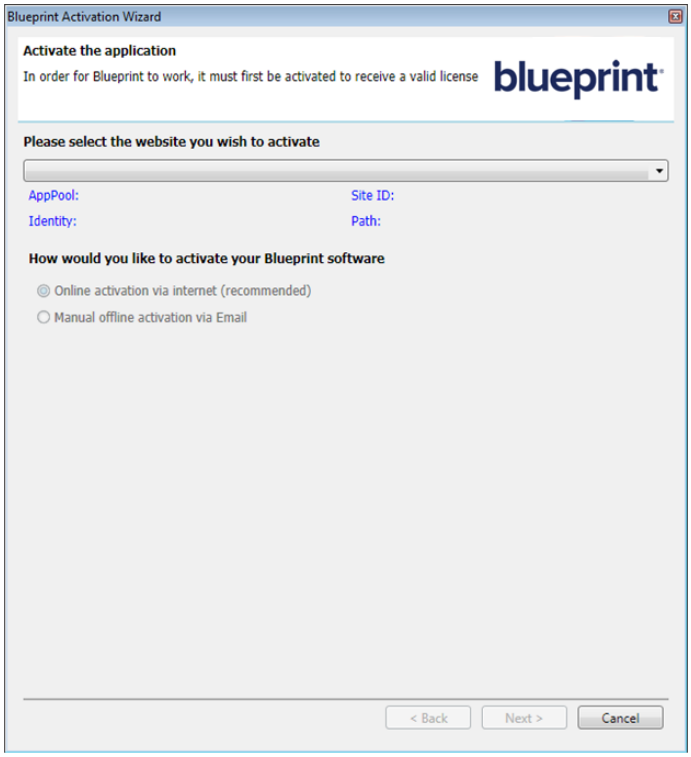
Complete the following steps to activate your Blueprint software online:

1. Launch the Blueprint Activation Wizard.

   The **`BlueprintActivationWizard.exe`** file is located in a folder called **`Setup`**. The **`Setup`** folder is located in the directory that you chose for installing Blueprint. For example, the default path is:

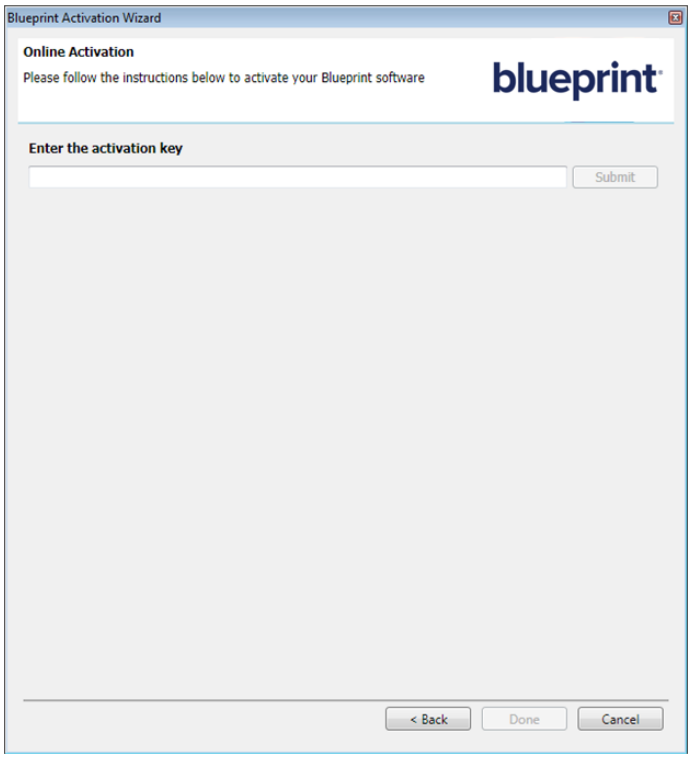
   `C:\Program Files\Blueprint Software Systems\Blueprint\Setup`

   The *Blueprint Activation Wizard* looks like this:



2. Select the website you want to activate by selecting an option from the drop-down list. The drop-down list only displays Blueprint sites.
3. Select the **Online activation via internet** option.
4. Click **Next**.

   The *Online Activation* screen appears:



5. Type your activation key into the space provided.
6. Click **Submit**.

   A Success dialog appears after Blueprint is activated. Click **OK** to close the dialog.

7. Click **Done** to close the Blueprint Activation Wizard.

To verify that Blueprint was activated successfully, navigate to the Blueprint URL (example: http://localhost:8080/). If the software was activated successfully, you will see a login screen instead of the error message.

Continue to Step 4: Blueprint instance setup to setup the Blueprint instance.

Activating Blueprint using manual offline activation

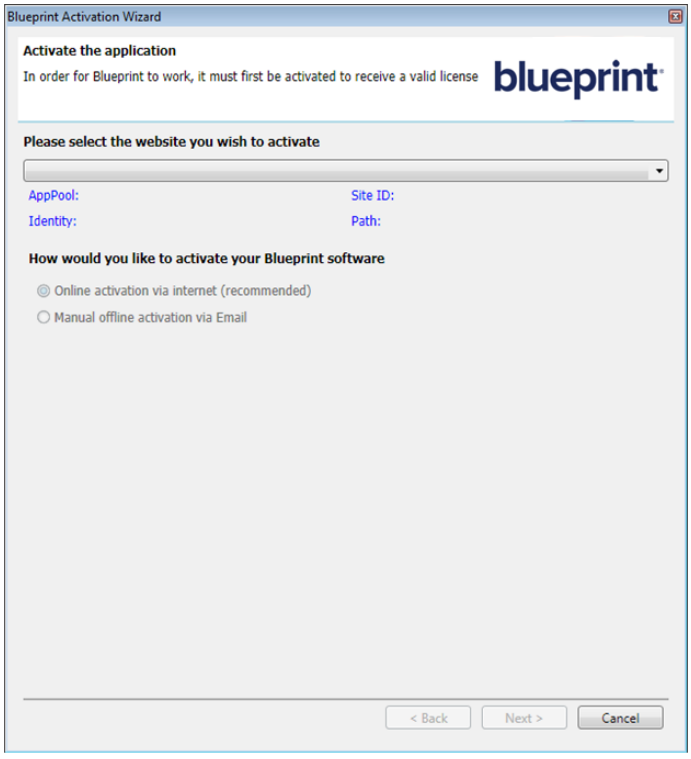
Complete the following steps to activate your Blueprint software offline:

1. Launch the Blueprint Activation Wizard.

   The **`BlueprintActivationWizard.exe`** file is located in a folder called **`Setup`**. The **`Setup`** folder is located in the directory that you chose for installing Blueprint. For example, the default path is:

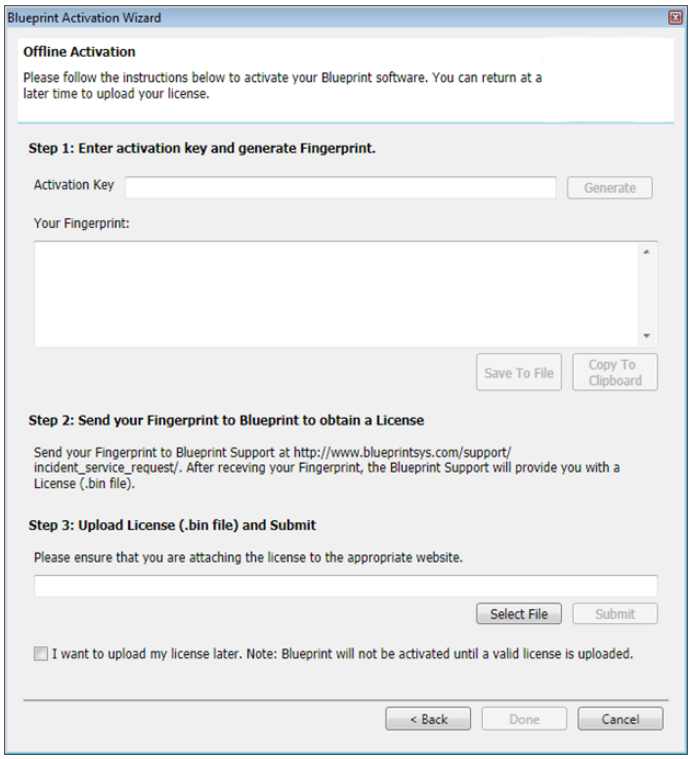
   `C:\Program Files\Blueprint Software Systems\Blueprint\Setup`

   The *Blueprint Activation Wizard* looks like this:



2. Select the website you want to activate by selecting an option from the drop-down list. The drop-down list only displays Blueprint sites.
3. Select the **Manual offline activation via Email** option.
4. Click **Next**.

   The *Offline Activation* screen appears:



5. Type your activation key into the space provided.
6. Click **Generate**.

   A fingerprint is generated and displayed in the **Your fingerprint** field.
7. Send the generated fingerprint to Blueprint Support at http://www.blueprintsys.com/support/incident_service_request/

   > **Tip**: Click **Save To File** or **Copy To Clipboard** so you don't have to type out the fingerprint manually.

   The Blueprint support team will provide you with a license (`.bin` file) after receiving your fingerprint. To close the Activation Wizard while waiting for your license file, click the **I want to upload my license later** option and then click **Done**.

   > **Note**: Blueprint is not activated yet.
8. After you have received your license (`.bin` file) from the Blueprint support team via Email, you must return to the *Offline Activation* screen to complete the activation. To return to the *Offline Activation* screen, launch the Blueprint Activation Wizard, select a Blueprint site, select the **Manual offline activation via Email** option, and then click **Next**.
9. Click **Select File**.
10. Locate and select the `.bin` license file.
11. Click **Submit**.

    A Success dialog appears after Blueprint is activated. Click **OK** to close the dialog.
12. Click **Done** to close the Blueprint Activation Wizard.

To verify that Blueprint was activated successfully, navigate to the Blueprint URL (example: http://localhost:8080/). If the software was activated successfully, you will see a login screen instead of the error message.

Continue to Step 4 to setup the Blueprint instance.

Step 4: Blueprint instance setup

Complete the following steps to setup your Blueprint instance:

1. Login to Blueprint:
    1. Open your browser and navigate to the URL where Blueprint is installed.
       Example: `http://localhost:8080`
       If the installation and configuration was successful, the login page is displayed.
    2. Enter the following credentials and click **Login**.
        - Login: admin
        - Password: changeme
    3. Read the EULA and click **I Agree** if you agree to the terms and conditions.
       You cannot continue until you accept the EULA.
2. Change the **admin** account password.
    1. Open the *Instance Administration Console* by clicking the application menu Menu and then clicking **Manage** > **Access the Instance Administration Console**.
    2. Open the *Users* tab by clicking **Manage Users And Groups** > **Users** on the ribbon.
    3. Click the **admin** user in the table.
       The user details are displayed in the panel on the right of the window.
    4. Click the **Change Password** button located at the bottom of the panel on the right side of the window.
    5. Type the new password in the appropriate fields and click **OK**.
    6. Click **Save**.
    7. Remain in the Instance Admin Console, and proceed to the next step.
3. Create a new user for administration purposes and assign instance admin rights to the new user:

    Note: We recommend creating a new user account for administration purposes so the **admin** account remains intact as a backup administration account. For example, if you create a new account called **administrator**, and then later accidentally revoke instance administration privileges from the **administrator** account, you can reapply those permissions using the **admin** account.

    1. Create a new Database user by clicking **New** > **New Database User** on the ribbon.
       A new user (<New User>) is added to the table.
    2. On the rightmost panel, enter the information for your new administrator user.

        Important: Ensure that you place a checkmark beside the **Instance Admin** option.

    3. Click **Save**.
4. Logout of Blueprint using the **Logout** link located in the upper right corner of the application.
5. Login to Blueprint using the new user that you created (example: administrator).
6. (Optional) Setup active directory integration. Refer to the *Instance Administration Guide* for more information.

    To configure default active directory integration:

    1. Open the *Instance Administration Console*.
    2. Click **Active Directory Settings**.

3. Select the **Enable Active Directory Integration** option.
4. Select the **Use default connection on identity** option.

> **Note**: The default connection only works if your Blueprint Server User (example: **`acme\rrunner`**) is a member of the active directory and the Blueprint Application Server is also a member of the active directory.

5. Click **Save**.

To configure custom active directory integration:

1. Open the *Instance Administration Console*.
2. Click **Active Directory Settings**.
3. Select the **Enable Active Directory Integration** option.
4. Select the **Use custom Active Directory integration** option.
5. Click the **Add** button.
6. Specify the active directory information on the rightmost side of the screen:
   - **Setting Name**: Choose a name for this active directory server so you can easily identify it in the list.
   - **Bind User**: Defines the user name of a user that has access to read from the active directory server. This user name must be the SamAccountName of the Bind User (not the common name, as per RC2010).

     > Note: The Bind User must be specified like this: **[DomainName]\[UserName]**. Example: **`BPTEST\root`**

   - **Bind Password**: Defines the password of the Bind User.
   - **Active Directory Authentication URL**: Defines the authentication URL of the active directory server. Example: **`LDAP://bpsdc-neo.blueprint.toronto/DC=blueprint,DC=Toronto`**
7. Click **Save**.

7. (Optional) Setup federated authentication if you want to enable Blueprint's single sign-on capabilities.
8. (Optional) Setup the Email Settings (SMTP) if you want to enable notifications.
9. Create projects.
10. Add users to Blueprint.
11. Create license groups.
12. Grant access to projects by creating project role assignments.

Instance administration tips

- **Project Templates**: You may want to consider creating one or more new project(s) in a folder called Templates. Then, you can create new projects using those templates. Project templates provide an efficient way to setup project groups and roles once, rather than configuring the project roles and groups every time

you create a new project.

- **Access Rights**: Consider using groups when you create project role assignments instead of including individual users. For example, whenever a new user requires *write* access to the **Getting Started** project, all you need to do is add the new user to Blueprint and then add the user to the **All Authors** group. The new user will immediately have access to all projects that include the **All Authors** group in a project role assignment.

Step 5: Blueprint client setup

Configuring elevated trust in-browser

Blueprint must be configured to run with elevated trust in-browser before you can use some advanced features, such as:

- screen capture capabilities
- pasting images into diagrams
- Visio integration, such as importing and exporting diagrams
- rich text table integration with other applications

Elevated trust in-browser can be configured manually on each client machine, or the configurations can be pushed to Windows computers in a centralized manner.

Using group policy to push the elevated trust-in browser configurations to Windows computers in a centralized manner

The recommended way to configure Blueprint to run with elevated trust in-browser is to use Group Policy. Group Policy allows IT Administrators to push configurations to Windows computers in a centralized manner.

For overview information about Group Policy, refer to Microsoft's Group Policy documentation at: http://technet.microsoft.com/en-us/windowsserver/bb310732.aspx.

To configure Blueprint to use elevated trust in-browser, your Group Policy configuration must do the following:

1. Set one of the following registry values:
   - On 32-bit computers:
     Set the **`HKEY_LOCAL_MACHINE\Software\Microsoft\Silverlight\AllowElevatedTrustAppsInBrowser`** registry value to **0x00000001**.
   - On 64-bit computers:
     Set the **`HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Silverlight\AllowElevatedTrustAppsInBrowser`** registry value to **0x00000001**.

   To learn more about setting a registry value through Group Policy, please refer to the Microsoft documentation that explains how to configure a registry item at: http://technet.microsoft.com/en-us/library/cc753092.aspx.
2. Download the elevated trust in-browser package from the Blueprint Customer Portal.

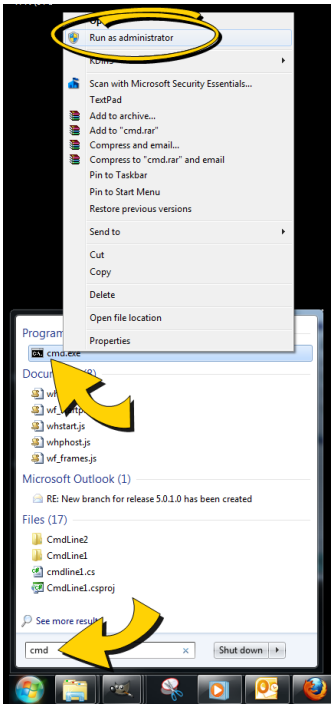
3. Add the **`publicBlueprintCertificate2017.cer`** certificate to the Trusted Publishers Store.

   To learn more about adding a certificate through Group Policy, please refer to the Microsoft documentation that explains how to deploy certificates by using group policy (http://technet.microsoft.com/en-us/library/cc770315(v=ws.10).aspx).

Manually configuring a computer to run with elevated trust in-browser

To configure Blueprint to run with elevated trust in-browser, perform the following steps on each client:

1. Download the elevated trust in-browser configuration files.
2. Unzip the package and note the directory where the files are located.
3. Run **`cmd.exe`** as Administrator.
   1. Click the Windows **Start** menu and type **cmd.exe** into the search bar.
   2. Right-click the **cmd.exe** program that appears under the Programs heading and then select **Run as administrator**:



   3. When the confirmation dialog appears, click **Yes**.

      The **`cmd.exe`** application launches with Administrator privileges:

4. Use the **`cd`** command to navigate to the folder where you unzipped the files.

   For example:

   ```
   cd c:\temp\elevated_trust
   ```

5. Enter the following commands to allow elevated trust to run on your local machine:

- For 64-bit operating systems:

```
regedit.exe /s AllowElevatedTrustAppsInBrowser64.reg
```

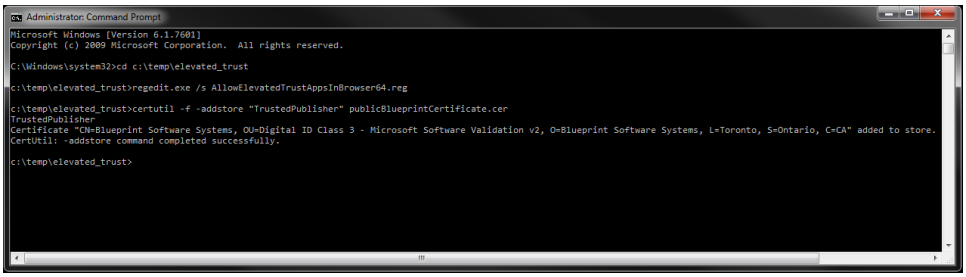
- For 32-bit operating systems:

```
regedit.exe /s AllowElevatedTrustAppsInBrowser.reg
```

6. Run the following `certutil` command to apply the Blueprint public certificate:

```
certutil.exe -f -addstore "TrustedPublisher"
publicBlueprintCertificate2017.cer
```

Here is an example of the commands run on a 64-bit operating system:



7. Restart your web browser for the changes to take effect.

Appendix

Configuration utility command reference

Tip

You can type the following commands to view more information about the command parameters:

```
blueprintwebcmd.exe /help
blueprintdbcmd.exe /help
```

Web Application Server Configuration Parameters

Parameter	Description	Default	Example
/object	Defines the object type of the command. This parameter can be set to one of the following values: ■ SITE ■ APPPOOL ■ DBCONFIG		
/command	Defines the command to perform. This parameter can be set to one of the following values: ■ LIST ■ ADD ■ DELETE ■ START ■ STOP		
/wsname	Defines the name of the site. This should be the same as the application pool name.	Blueprint	Blueprint
/wsid	Defines the ID of the site.		25
/port	Defines the port number used for the site.		8080
/dir	Defines the location of the Blueprint installation.		C:\Program Files\Blueprint Software Systems\Blueprint\Web
/apppoolname	Defines the name of the application pool. This should be the same as the site name.	Blueprint	Blueprint
/datasource	Defines your database and instance names.		DBSERVER\INSTANCE01
/catalog	Defines the name of the database.	Blueprint	

Parameter	Description	Default	Example
/integratedsec	Defines whether or not Windows security is used. This parameter can be set to one of the following values: ■ TRUE ■ FALSE If /integratedsec is set to FALSE, you must specify a /userid and /password.		
/userid	Defines the username of the Service Account/Application Pool user.		
/password	Defines the password of the Service Account/Application Pool user.		

Database Server Configuration Parameters

Parameter	Description	Default	Example
/object	Defines the object type of the command. This parameter can be set to one of the following values: ■ SERVER ■ DB ■ USER		
/command	Defines the command to perform. This parameter can be set to one of the following values: ■ LIST ■ ADD ■ INIT ■ UPGRADE		
/datasource	Defines your database and instance names.		DBSERVER\INSTANCE01
/catalog	Defines the name of the database.	Blueprint	BlueprintDB
/integratedsec	Defines whether or not Windows security is used. This parameter can be set to one of the following values: ■ TRUE ■ FALSE If /integratedsec is set to FALSE, you must specify a /userid and /password.		
/userid	Defines the username of the *Database System Administrator* user. This parameter is only required if /integratedsec is set to FALSE.		
/password	Defines the password of the *Database System Administrator* user. This parameter is only required if /integratedsec is set to FALSE.		
/nuseridentity	Defines the username of the *Blueprint Server User*.		acme\rrunner

Maintaining the Blueprint database

For best performance, we recommend that you perform routine maintenance on the Blueprint database. For more information, login to the Blueprint Customer Portal (http://portal.blueprintsys.com) and refer to *Knowledge Base Article 1046, How to perform routine maintenance on the Blueprint database*.

Setting up a new database

In certain cases, a new database may be need to be set up. For more information on setting up a new database, see the "Setting up a new database" section in the *IT Administration Guide*.

Setting up federated authentication

Refer to the *Instance Administration Guide* for more information.

To configure your identity provider for Blueprint federated authentication, ensure the following requirements are met:

- The **Entity ID** must be set to:

  ```
  <Blueprint_URL>/Login/SAMLHandler.ashx
  ```

  where <Blueprint_URL> is your main Blueprint URL.

  > **Example**
  > For Blueprint cloud customers, the **Entity ID** will look something like this:
  > **`https://acme.blueprintcloud.com/Login/SAMLHandler.ashx`**
  > For Blueprint on-premise customers, the **Entity ID** will look something like this:
  > **`https://blueprint.acme.com/Login/SAMLHandler.ashx`**

- The **POST Endpoint** must be set to:

  ```
  <Blueprint_URL>/Login/SAMLHandler.ashx
  ```

  where <Blueprint_URL> is your main Blueprint URL.

- A **Username** attribute must be included in the SAML response (that is, the token).

  Blueprint reads the username from the **Username** attribute in the token (not the Subject). The name of this attribute must be **Username**. The username can be in format you want, but must match the usernames as created in Blueprint. Valid options are regular usernames, Windows/AD account names (DOMAIN\user), e-mail addresses, Distinguished Names, or x509 Subjects.

- The SAML response must contain the identity provider certificate (x509).

To enable Blueprint federated authentication:

1. Open the *Instance Administration Console*.
2. Click **Federated Authentication**.
3. Select the **Enable Federated Authentication** option.
4. Set your federated authentication settings:
    - Click **Replace** to upload your Identity Provider Certificate. The certificate must be in DER format.

      > **Important**: Certificates have an expiry date. Make sure you replace your certificate before it expires or users will be unable to access Blueprint.

    - **Login URL**: Defines your Identity Provider Login Service URL. This is the URL that Blueprint navigates to when the user clicks the Go button on the login screen. At this time, the Identity Provider returns a authentication token to Blueprint to authenticate the user.

      Example: https://idp.domain.com/adfs/ls/
    - **Logout URL**: Defines the URL to navigate to after a user clicks the Logout button in Blueprint. This behavior is not applicable if a user is logged in with fallback authentication.
    - **Error URL (optional)**: If a token error occurs, the user is redirected to the specified URL. The specific error is included as a GET parameter in the URL.

      If an **Error URL** is not provided, Blueprint displays the token errors in the popup window.
    - **Login Prompt (optional)**: Defines the login text that appears on the login screen when Federated Authentication is enabled:

      The default text is:

      Login with Corporate Credentials
5. Click **Save**.

Setting up email notifications

E-mail settings are required in Blueprint if you want to take advantage of Blueprint notifications. Blueprint notifications provide your users with information and reminders at key moments. Notifications can help users stay up-to-date with project developments. There are two types of notifications you can enable: review notifications and email integrated discussions notifications.

Refer to the *Instance Administration Guide* for more information.

Setting up review notifications

Perform the following steps if you want to enable review notifications and configure the associated settings:

1. Open the *Instance Administration Console*.
2. Click **Configure Instance** > **E-mail Settings** on the ribbon (*Instance Admin* tab, *Instance* group).
3. Select the **Enable Review Notifications** check box to enable review notifications.
4. Enter your e-mail credentials in the *Email Credentials* section:
    - **E-mail Address**: Defines the e-mail address that will appear in the **From** address for all e-mail notifications.
    - **User Name**: Defines the user name of the e-mail account.
    - **Password**: Defines the password of the user.
5. Enter your outgoing mail server settings and preferences:
    - **Server IP / Hostname**: Defines the IP address or hostname of your SMTP server.
    - **Port**: Defines the port number of your SMTP server.
    - **Enable SSL**: Defines whether or not the SMTP server requires SSL.
    - **Authenticated SMTP**: Defines whether or not SMTP authentication is required. If authentication is required, select this option and enter a valid user name and password.
        - **User Name**: Defines the user name of a user with access to the SMTP server. This user name can be different from the user name provided in the *Email Credentials* section.

          Note: The SMTP user name is sometimes, but not always, the e-mail address of the user. The format of the user name is dependent on the server configuration.
        - **Password**: Defines the password of the user.
6. Click **Save**.

Tip: You can click the **Send Test E-mail** button to verify that e-mails can be sent successfully.

Setting up e-mail integrated discussions

Perform the following steps to enable and configure e-mail integrated discussion settings:

1. Open the *Instance Administration Console*.
2. Click **Configure Instance** > **E-mail Settings** on the ribbon (*Instance Admin* tab, *Instance* group).
3. Select the **Allow projects to enable discussions via E-mail** check box to enable e-mail-integrated discussions.

> **Note:** By default, e-mail integrated discussions are set to only allow users to mention Blueprint registered users.
>
> To change this setting: click **Edit Settings**. Next, click **All users** to allow any user outside of Blueprint to contribute via e-mail to discussions.
>
> To restrict this setting to a subset of e-mail domains: ensure **Specify domains** is enabled, enter the domains you want to allow in e-mail integrated discussions and click **OK**.

4. Enter your e-mail credentials in the *Email Credentials* section:
    - **E-mail Address**: Defines the e-mail address that will appear in the **From** address for all e-mail notifications.
    - **User Name**: Defines the user name of the e-mail account.
    - **Password**: Defines the password of the user.
5. Enter your incoming mail server settings and preferences:
    - **IMAP/POP**: Defines the protocol of the incoming email server.
    - **Server IP/Hostname**: Defines the IP address or hostname of your IMAP/POP server.
    - **Port**: Defines the port number of your IMAP/POP server.
    - **Enable SSL**: Defines whether or not the IMAP/POP server requires SSL.

    > **Tip**: You can click the **Test Connection** button to verify that e-mail integrated discussions can be delivered successfully.
6. Enter your outgoing mail server settings and preferences:
    - **Server IP / Hostname**: Defines the IP address or hostname of your SMTP server.
    - **Port**: Defines the port number of your SMTP server.
    - **Enable SSL**: Defines whether or not the SMTP server requires SSL.
    - **Authenticated SMTP**: Defines whether or not SMTP authentication is required. If authentication is required, select this option and enter a valid user name and password.
        - **User Name**: Defines the user name of a user with access to the SMTP server. This user name can be different from the user name provided in the *Email Credentials* section.

        > **Note**: The SMTP user name is sometimes, but not always, the e-mail address of the user. The format of the user name is dependent on the server configuration.

        - **Password**: Defines the password of the user.
7. Click **Save**.

Next, enable the e-mail integrated discussions setting within Project Settings.

Adding users to Blueprint

Refer to the *Instance Administration Guide* for more information.

- Complete the following steps to add all Active Directory users to Blueprint:

  Important: You can only add Windows users if Active Directory integration is enabled.

  - Click **Manage Users And Groups** > **Users** on the ribbon (*Instance Admin* tab, *Instance* group).
  - Click **New** > **New Windows User** on the ribbon (*Instance Admin* tab, *Manage Items* group).
  - Click the **Find** button to display all Active Directory users.

    Note: If Active Directory integration is enabled, the Location is automatically populated so you can access the Active Directory.

  - Type **`Ctrl-a`** to select all users and then click **OK**.
- Complete the following steps to add a single database user to Blueprint:
  - On the *Users* tab, click **New** > **New Database User** on the ribbon (*Instance Admin* tab, *Manage Items* group).
  - Enter the user information on the right side of the window.
  - Click **Save**.

Creating license groups

Refer to the *Instance Administration Guide* for more information.

A license group is an instance-level group that allows you to control the type of license that a user consumes while logged into Blueprint. A user's effective access in Blueprint is the intersection of their project role assignment and their license.

Important: Users must be added to an *Author* or *Collaborate* license group before they can take advantage of most Blueprint features. Users that are not added to an *Author* or *Collaborate* license group are limited to accessing Blueprint artifacts by URL.

Complete the following steps to create an **All Authors** group that is designated as an *Author* license group:

1. Click **Manage Users And Groups** > **Groups** on the ribbon (*Instance Admin* tab, *Instance* group).
2. Click **New** > **Database Group** on the ribbon (*Instance Admin* tab, *Manage Items* group).
3. Enter the group information:
   - **Name**: Set this field to **All Authors**.
   - **Description**: Specify a description for the group.
   - **Email**: Specify an email address for the group.
   - **Scope**: This field must be left blank. License groups cannot have an associated Scope.
   - **License Group?**: Enable this option so the group is a license group.
   - **License Type**: Change this option to **Author**.
4. Click the **Add** button to add members to the group. Type **`Ctrl-a`** to select all users, and then click **OK**.
5. Click **Save**.

Repeat the steps above to create an **All Contributors** group, but set the **License Type** to **Collaborate**.

Creating projects

Refer to the *Instance Administration Guide* for more information.

Complete the following steps to create the **Getting Started** project.

> **Note**: The purpose of the Getting Started project is to provide a location for users to experiment with Blueprint features and complete the tutorials in the Getting Started Guide.

1. Click the **Projects** button on the ribbon.
2. Right-click the *Blueprint* item on the left side of the window and select **New Folder**.
3. Specify the following folder information:
   - **Name**: Getting Started
   - **Description**: Folder containing getting started project(s).
4. Click **Save**.
5. Expand the *Blueprint* item on the left side of the window, right-click **Getting Started**, and select **New Project**.
6. Specify the project information:
   - **Name**: Getting Started
   - **Description**: Project for users to learn Blueprint using the Blueprint Getting Started Guide.
   - **Location**: This should be set to /Blueprint/Getting Started/
   - **Select Source**: Empty Project
7. Click **Save**.
8. Click the **Launch Project Administration** button. This button is located in the lower-right area of the window. The *Project Administration Console* is displayed.

Granting access to projects

Refer to the *Project Administration Guide* for more information.

> **Note**: A user's effective access in Blueprint is the intersection of their project role assignment and their license.

Complete the following steps to configure the **Getting Started** project so the **All Authors** group can modify the project.

1. Create an Authors role.
   1. In the *Project Admin Console*, click **Manage Access** > **Project Roles** on the ribbon (*Project Admin* tab, *Project* group)
   2. Click the **New** button on the ribbon (*Project Admin* tab, *Actions* group).
   3. Specify the role information:
      - **Name**: Authors
      - **Description**: This role has read, edit, trace, and comment privileges.

- **Privileges**: Place a checkmark beside **Read**, **Edit**, **Trace**, and **Comment**.

4. Click **Save**.

2. Assign the **Authors** role to the **All Authors** group for the **Getting Started** project.

    1. In the *Project Admin Console*, click **Manage Access** > **Project Role Assignments** on the ribbon (*Project Admin* tab, *Project* group)
    2. Click the **New** button on the ribbon (*Project Admin* tab, *Actions* group) and then click the *Groups* tab after the dialog appears.
    3. Select the **All Authors** group and click **OK**.
    4. Specify the project role assignment information:

        - **Identity**: Group : All Authors
        - **Role**: Authors
        - **Scope**: Project

    5. Click **Save**.

Web.config reference

> **Note:** Before upgrading Blueprint, we recommend backing up the **`web.config`** file. None of your existing customizations are preserved during the upgrade process. The backup file can be used as a reference to make changes to the new version of **`web.config`** after upgrading.

The **`web.config`** file contains many Blueprint web application settings, most of which should not be edited in common configuration scenarios. The following table identifies Blueprint settings that are commonly configured as well as their possible values:

Section	Parameter name	Values	Description
<system.web> <authentication> <forms>	loginUrl	Login/WinLogin.aspx	The default value (Login/WinLogin.aspx) causes a Windows log-in dialog to appear after Blueprint is opened. You have the option of removing the Windows dialog by changing this value. **Caution:** If the value is changed to weblogin.aspx, users will no longer be able to log on using LDAP credentials. Only Blueprint database users will be permitted to log on. **Tip:** If you change the loginUrl value, you can test the results by opening Blueprint in a browser.
<appSettings> <add key="LdapGetTimeout">	value	300	Indicates how many seconds it takes for LDAP retrieval to time out. By default, the LDAP retrieval timeout is 300 seconds (recommended). The value can be changed. The minimum value it can be changed to is 30 and the maximum is 600.

Section	Parameter name	Values	Description
<appSettings> <add key="UseLegacyDomainName">	value	FALSE	The UseLegacyDomainName key indicates the domain name section. You can use the original parsing algorithm to determine the domain name. There is no need to change the default value in most circumstances. However, if you have problems integrating Blueprint with LDAP, you may need to change the value to TRUE. Please contact Blueprint Support before changing this value.
<appSettings> <add key="AttachmentsFolderPath">	value	""	When Blueprint users save attachments to their system, the attachments get saved to their IIS Application Pool Temp folder by default. To store the attachment in another location, replace the empty value with the desired path.
<appSettings> <add key="UserSessionExpirationTimeoutInMinutes">	value	30	This section defines when the session times out from inactivity and, as a result, the user is logged off of Blueprint. The value is measured in minutes. The default session timeout is 30 minutes. The recommended value is 20. The minimum value it can be changed to is 10 and the maximum is 60.

Section	Parameter name	Values	Description
<appSettings> <add key="QCMaxRetries">	value	30	This section defines the maximum number of connection re-attempts that will be made when connection to HP Quality Center initially fails. By default, the value is 30 (recommended). To change the value, the following tag must be manually added to the <connectionStrings> section first: <add key="QCMaxRetries" value="">"> The minimum value it can be changed to is 3 and the maximum is 60.
<appSettings> <add key="QCDelayInSeconds">	value	30	This section defines the delay between connection re-attempts that will be made when connection to HP Quality Center initially fails. The delay is measured in seconds. To change the value, the following tag must be manually added to the <connectionStrings> section first: <add key="QCDelayInSeconds" value="">"> The recommended value is 5. The minimum value it can be changed to is 5 and the maximum is 60.

Section	Parameter name	Values	Description
<appSettings> <add key="ExcelImportArtifactLimit">	value	500	This section defines the amount of artifacts that a user can import from Microsoft Excel at a time. **Note:** We do not recommend changing this value unless it is necessary.
<appSettings> <add key="ConnectionCheckInterval">	value	2	This section defines the amount of minutes after which Silverlight checks if the user session on the server is expired. This interval triggers session renewal. **Important:** The `ConnectionCheckInterval` value should always be less than the `sessionState` value (timeout).
<connectionStrings> <add>	DataSource	<dbserver>,<dbport>	You can use this section to point the Blueprint instance to a different database instance. Before specifying valid values for the database server and the database port, this needs to be manually added to the `<connectionStrings>` section first: `<add key="DataSource" value="">`
<connectionStrings> <add>	Initial Catalog	<db>	Indicates the name of the Blueprint database. By default, the name is `Blueprint`. Before changing the database name, this needs to be manually added to the `<connectionStrings>` first: `<add key="InitialCatalog" value="NewDatabaseName">`

Installing services manually

The following services are available to install if needed:

- **Job services (recommended)**

  This functionality is necessary to perform the following jobs in Blueprint: document generation, exporting artifacts to ALM systems and test generation.

- **Legacy support for HP Quality Center versions 12 and earlier (optional)**

  If you require support for HP Quality Center version 12 or earlier, setup of this component is necessary.

  > **Note:** HP Quality Center support is only available for COM library. It is not available for the REST API.

To install services:

- Continue to the single-server section to host job services and the Blueprint database on the same server.
- Continue to the distributed-server section to host job services and the Blueprint database on separate servers.

Setting up services (single-server setup)

You can install one or both of the following:

- 64-bit job services
- HP Quality Center legacy support

Deploying 64-bit job services (single-server)

1. Copy the following folder: **`C:\Program Files (x86)\Blueprint Software Systems\Blueprint\JobExecutorService\bin`**.
2. Paste the copied folder into the following directory: **`C:\Program Files (x86)\Blueprint Software Systems\Blueprint\JobExecutorService`**.
3. Rename the pasted folder **64bitservice**.
4. Delete the following two files from the pasted folder: **BluePrintSys.RC.JobExecutor32.exe** and **BluePrintSys.RC.JobExecutor32.exe.config**.
5. Open the following configuration file in a text editor: **BluePrintSys.RC.JobExecutor.exe.config**
6. Make sure that the value in the `<add key="Service.Jobs">` tag is as follows:

```
<add key="Service.Jobs" value="
DocGen,TfsExport,HpAlmRestExport,TfsChangeSummary,HpAlmRestChangeSumma
ry,TfsExportTests,HpAlmRestExportTests" />
```

7. Make sure the `<add key="Service.Name">` tag specifies the 64-bit job service as follows:

```
<add key="Service.Name" value="Blueprint Job Service (64 bit)" />
```

8. Replace the following `connectionString` value with the Blueprint database connection string:

```
<connectionStrings>
<add name="InstanceContainer"
connectionString="metadata=res://*/Models.Instance.csdl|res://*/Models
.Instance.ssdl|res://*/Models.Instance.msl;provider=System.Data.SqlCli
ent;provider connection string=&quot;Data Source=.\MSSQLSERVER;Initial
Catalog=Blueprint;Integrated
Security=True;Pooling=False;MultipleActiveResultSets=True&quot;"
providerName="System.Data.EntityClient" />
</connectionStrings>
```

> **Note:** `Data Source` must specify the SQL instance name and the Blueprint instance name. If your SQL instance has a name that is different from **`MSSQLSERVER`** and/or your Blueprint instance is not named **`Blueprint`**, you need to change the value(s).

9. Repeat all of the above steps to deploy a second 64-bit job service, performing the following revisions:
   - Name the second copied and pasted folder as follows: **64bitservice2**
   - Specify the `Service.Name` value as follows: `Blueprint Job Service 2 (64 bit).`

   a. Install both 64-bit job services by running the following command with your user name and password:

```
BluePrintSys.RC.JobExecutor.exe -c Install -a [USER] -u
[USERNAME] -p [PASSWORD]
```

> **Note:** To install the service using the default Windows account, run the following command instead:
>
> `BluePrintSys.RC.JobExecutor.exe -c Install -a LocalService`

   b. Start the 64-bit job services with the following command:

```
BluePrintSys.RC.JobExecutor64.exe -c Start
```

You have successfully deployed the 64-bit job services.

Setting up HP Quality Center legacy support (single-server)

Installing HP Quality Center legacy support involves completing the following steps:

- Step One: Setting up the HP Quality Center legacy support connector
- Step Two: Deploying 32-bit job services for HP QC legacy support

Step One: Setting up the HP Quality Center legacy support connector

1. Set up the HP QC application pool by running the following command:

```
blueprintqcwebcmd.exe  /object AppPool /command ADD /apppoolname
HPQCLegacyConnector
```

2. Set up the HP QC web site by running the following command (where the number after `/port` is `QcLiteWeb`'s port number):

```
blueprintqcwebcmd.exe  /object Site /command ADD /wsname
HPQCLegacyConnector /dir "C:\Program Files (x86)\Blueprint Software
Systems\Blueprint\QcLiteWeb" /port [8081] /apppoolname
HPQCLegacyConnector
```

3. Start the HP QC application pool by running the following command:

```
blueprintqcwebcmd.exe  /object AppPool /command START /apppoolname
HPQCLegacyConnector
```

4. Start the HP QC web site by running the following command:

```
blueprintqcwebcmd.exe  /object SITE /command START /wsname
HPQCLegacyConnector
```

5. Set the HP QC key for Blueprint with the following command (where the number after `/port` is `QcLiteWeb`'s port number):

```
blueprintqcwebcmd.exe /object Config /command SET /dir "C:\Program
Files (x86)\Blueprint Software Systems\Blueprint\Web" /port [8081]
```

You have successfully set up the HP QC legacy support connector.

Step Two: Deploying 32-bit job services for HP QC legacy support

1. Copy the following folder: **`C:\Program Files (x86)\Blueprint Software Systems\Blueprint\JobExecutorService\bin`**.
2. Paste the copied folder into the following directory: **`C:\Program Files (x86)\Blueprint Software Systems\Blueprint\JobExecutorService`**.
3. Rename the pasted folder **32bitservice** for quick reference.
4. Delete the following two files from the pasted folder: **BluePrintSys.RC.JobExecutor.exe** and **BluePrintSys.RC.JobExecutor.exe.config**.
5. Open the following configuration file in a text editor: **BluePrintSys.RC.JobExecutor32.exe.config**
6. Make sure that the value within `<add key="Service.Jobs">` is as follows:

```
<add key="Service.Jobs" value="QcExport,QcChangeSummary,QcExportTests"
/>
```

7. Make sure the `<add key="Service.Name">` tag includes the Blueprint HP QC legacy job service as follows:

```
<add key="Service.Name" value="Blueprint HP QC Legacy Job Service (32
bit)" />
```

8. Replace the following `connectionString` value with the Blueprint database connection string:

```
<connectionStrings>
<add name="InstanceContainer"
connectionString="metadata=res://*/Models.Instance.csdl|res://*/Models
.Instance.ssdl|res://*/Models.Instance.msl;provider=System.Data.SqlCli
ent;provider connection string=&quot;Data Source=.\MSSQLSERVER;Initial
Catalog=Blueprint;Integrated
Security=True;Pooling=False;MultipleActiveResultSets=True&quot;"
providerName="System.Data. EntityClient" />
</connectionStrings>
```

> **Note:** `Data Source` must specify the SQL instance name and the Blueprint instance name. If your SQL instance has a name that is different from **`MSSQLSERVER`** and/or your Blueprint instance is not named **`Blueprint`**, you need to change the value(s).

   a. Install the 32-bit service by running the following command with your user name and password:

```
BluePrintSys.RC.JobExecutor32.exe -c Install -a [USER] -u
[USERNAME] -p [PASSWORD]
```

> **Note:** To install the service using the default Windows account, run the following command instead:
>
> `BluePrintSys.RC.JobExecutor32.exe -c Install -a LocalService`

   b. Next, start the 32-bit job executor Windows service with the following command:

```
BluePrintSys.RC.JobExecutor32.exe -c Start
```

You have successfully installed legacy support for HP Quality Center.

Setting up services (distributed-server setup)

To set up services on a separate machine (distributed setup), complete the following steps:

1. Deploy services
2. Test the connection to the database
3. Finalize the job services setup

Step One: Deploying services

You have the option of setting up the following services:

- 64-bit job services
- HP Quality Center legacy support

Deploying the 64-bit services (distributed-server)

Note: In certain cases, the job services folder and files mentioned in the instructions below are located in the **C:\Program Files** directory instead of the **C:\Program Files (x86)** directory.

1. Copy the following folder: **C:\Program Files (x86)\Blueprint Software Systems\Blueprint\JobExecutorService\bin**.
2. Paste the copied folder into the following directory: **C:\Program Files (x86)\Blueprint Software Systems\Blueprint\JobExecutorService**.
3. Rename the pasted folder **64bitservice**.
4. Delete the following two files from the pasted folder: **BluePrintSys.RC.JobExecutor32.exe** and **BluePrintSys.RC.JobExecutor32.exe.config**.
5. Open the following configuration file in a text editor: **BluePrintSys.RC.JobExecutor.exe.config**
6. Make sure that the value in the `<add key="Service.Jobs">` tag is as follows:

```
<add key="Service.Jobs" value="
DocGen,TfsExport,HpAlmRestExport,TfsChangeSummary,HpAlmRestChangeSummary,TfsExportTests,HpAlmRestExportTests" />
```

7. Make sure the `<add key="Service.Name">` tag specifies the 64-bit job service as follows:

```
<add key="Service.Name" value="Blueprint Job Service (64 bit)" />
```

8. Replace the following `connectionString` value with the Blueprint database connection string:

```
<connectionStrings>
<add name="InstanceContainer"
connectionString="metadata=res://*/Models.Instance.csdl|res://*/Models.Instance.ssdl|res://*/Models.Instance.msl;provider=System.Data.SqlClient;provider connection string=&amp;quot;Data Source=.\MSSQLSERVER;Initial Catalog=Blueprint;Integrated Security=True;Pooling=False;MultipleActiveResultSets=True&amp;quot;"
providerName="System.Data.EntityClient" />
</connectionStrings>
```

> **Note:** `Data Source` must specify the SQL instance name and the Blueprint instance name. If your SQL instance has a name that is different from **`MSSQLSERVER`** and/or your Blueprint instance is not named **`Blueprint`**, you need to change the value(s).

9. Repeat all of the above steps to deploy a second 64-bit job service, performing the following revisions:
   - Name the second copied and pasted folder as follows: **64bitservice2**
   - Specify the `Service.Name` value as follows: `Blueprint Job Service 2 (64 bit).`

Setting up HP Quality Center legacy support (distributed-server)

Setting up legacy support for HP Quality Center involves the following steps:

- Step One: Setting up the HP Quality Center legacy support connector
- Step Two: Deploying 32-bit job services

STEP ONE: SETTING UP THE HP QUALITY CENTER LEGACY SUPPORT CONNECTOR

1. Set up the HP QC application pool by running the following command:

```
blueprintqcwebcmd.exe  /object AppPool /command ADD /apppoolname
HPQCLegacyConnector
```

2. Set up the HP QC web site by running the following command (where the number after `/port` is `QcLiteWeb`'s port number):

```
blueprintqcwebcmd.exe  /object Site /command ADD /wsname
HPQCLegacyConnector /dir "C:\Program Files (x86)\Blueprint Software
Systems\Blueprint\QcLiteWeb" /port [8081] /apppoolname
HPQCLegacyConnector
```

3. Start the HP QC application pool by running the following command:

```
blueprintqcwebcmd.exe  /object AppPool /command START /apppoolname
HPQCLegacyConnector
```

4. Start the HP QC web site by running the following command:

```
blueprintqcwebcmd.exe  /object SITE /command START /wsname
HPQCLegacyConnector
```

5. Set the HP QC key for Blueprint with the following command (where the number after `/port` is `QcLiteWeb`'s port number):

```
blueprintqcwebcmd.exe /object Config /command SET /dir "C:\Program
Files (x86)\Blueprint Software Systems\Blueprint\Web" /port [8081]
```

You have successfully set up the HP QC legacy support connector.

STEP TWO: DEPLOYING 32-BIT JOB SERVICES

Note: In certain cases, the job services folder and files mentioned in the instructions below are located in the **C:\Program Files** directory instead of the **C:\Program Files (x86)** directory.

Complete the following instructions, pasting the copied files onto your target machine:

1. Copy the following folder: **C:\Program Files (x86)\Blueprint Software Systems\Blueprint\JobExecutorService\bin**.
2. Paste the copied folder into the following directory: **C:\Program Files (x86)\Blueprint Software Systems\Blueprint\JobExecutorService**.
3. Rename the pasted folder **32bitservice** for quick reference.
4. Delete the following two files from the pasted folder: **BluePrintSys.RC.JobExecutor.exe** and **BluePrintSys.RC.JobExecutor.exe.config**.
5. Open the following configuration file in a text editor: **BluePrintSys.RC.JobExecutor32.exe.config**
6. Make sure that the value within `<add key="Service.Jobs">` is as follows:

```
<add key="Service.Jobs" value="QcExport,QcChangeSummary,QcExportTests"
/>
```

7. Make sure the `<add key="Service.Name">` tag includes the Blueprint HP QC legacy job service as follows:

```
<add key="Service.Name" value="Blueprint HP QC Legacy Job Service (32
bit)" />
```

8. Replace the following `connectionString` value with the Blueprint database connection string:

```
<connectionStrings>
<add name="InstanceContainer"
connectionString="metadata=res://*/Models.Instance.csdl|res://*/Models
.Instance.ssdl|res://*/Models.Instance.msl;provider=System.Data.SqlCli
ent;provider connection string=&quot;Data Source=.\MSSQLSERVER;Initial
Catalog=Blueprint;Integrated
Security=True;Pooling=False;MultipleActiveResultSets=True&quot;"
providerName="System.Data. EntityClient" />
</connectionStrings>
```

Note: `Data Source` must specify the SQL instance name and the Blueprint instance name. If your SQL instance has a name that is different from **MSSQLSERVER** and/or your Blueprint instance is not named **Blueprint**, you need to change the value(s).

Step Two: Testing the connection to the database

> **Note:** This step must be performed for each job service you intend to set up.

This testing procedure must validate the configuration of the following values:

- The `Service.Name` value is unique and no other job service has this name on the current machine
- The `connectionString` value is valid and the job service can connect to Blueprint database.

If the test is not successful, you must specify the correct value(s) in the job service configuration file and re-attempt the test.

To test the job services connection to the Blueprint database:

- Run the following command (where `[JobServiceExecutableFile]` is the name of the job service executable file):

```
[JobServiceExecutableFile] -c test
```

> **Note:** The 32-bit job services file name is typically **BluePrintSys.RC.JobExecutor32.exe** and the 64-bit job services file name is typically **BluePrintSys.RC.JobExecutor.exe**.

Step Three: Finalizing the job services setup

> **Note:** This step must be performed for each job service you intend to set up.

This is the final step in setting up job services on a separate machine from the Blueprint database.

To finalize the setup:

1. Install the job service by running the following command (where `[JobServiceExecutableFile]` is the name of the job service executable file):

```
[JobServiceExecutableFile] -c install
```

2. Start the job service by running the following command (where `[JobServiceExecutableFile]` is the name of the job service executable file):

```
[JobServiceExecutableFile] -c start
```

You have deployed job services.

To verify whether job services have been successfully installed and configured:

1. Log on to Blueprint.
2. Open the *Instance Administration Console* from the **Menu** 

.

   The *Instance Administration Console* appears.

3. Click **Job Services**.

   The *Job Services* screen appears.

Any job services that have been successfully installed and configured appear in the *Job Services* list. Information about the configured service name and supported jobs also is available in the list.