Blueprint 6.3

Instance Administration Guide

Contents

Instance Administration

The *Instance Administration Console* allows you to create new projects and configure instance settings.

> **Note**: You must have instance admin privileges to access the *Instance Administration Console*.

About standard artifact types

A *standard artifact type* refers to a requirement classification in Blueprint. Standard artifact types are associated with all projects in the instance.

When creating a new standard artifact type, you can choose from a variety of templates (that is, base types) to model your standard artifact type from, such as:

- Textual requirement
- UI mockup
- Use case

Standard artifact types are also different from custom artifact types, which only apply to specific projects.

> **Example**
>
> Fatima, a business analyst, plans to configure multiple new projects that involve creating business and functional requirements in textual form. She wants to make sure there is a high level of clarity between business and functional requirements.
>
> Instead of creating these artifact types manually in every project, Fatima decides to leverage standard properties. At the instance level, Fatima decides to create two different standard artifact types called **Business Requirements** and **Functional Requirements**. She sets the `Base Type` to *TextualRequirement* for both of the new standard artifact types. Creating these standard artifact types allows her to standardize these new artifact types across projects. She can also rest assured knowing that project administrators will not be able to modify or reconfigure the properties at the project level.
>
> In any project, users can now choose a business requirement or a functional requirement when creating a new requirement to suit their needs.

Instance administrators with the [applicable privileges](#) can create and edit standard artifact types using the *Standard Artifact Types* tab in the *Instance Administration Console*. The *Standard Artifact Types* tab looks like this:

The left side of the page provides you with a table containing the following columns of information about each artifact type:

- *Name*: Indicates the name of the artifact type. This name appears in the list of options when users are selecting a new artifact to create. It also appears in the *Standard Artifact Type* column when users are viewing the artifact list.

- *Prefix*: Indicates the ID prefix of the artifact type. Prefixes are unique across artifact types. All artifacts have a unique ID that begins with this prefix.

- *Tooltip*: If specified, provides a tooltip description when users are selecting a new artifact to create and they hover over the artifact type.

- *Group Label*: If specified, this artifact type appears under the specified label when users are selecting a new artifact to create.

- *Base Type*: All artifact types must have a base type. The base type determines the type of editor that is used when a user opens an artifact. The available base types are pre-configured.

After you select a standard artifact type from the table, the standard artifact type details are displayed on the right side of the page.

Modifying standard artifact types

To modify a standard artifact type:

1. Open the *Standard Artifact Types* tab.

   1. Open the Instance Administration Console.

   2. Click the **Standard Artifact Types** link.

2. Select the artifact type you want to modify.

   Select an artifact type by clicking a row in the table. The artifact type details are displayed on the right side of the page.

3. Update the artifact type details.

   - `Name`: Indicates the name of the artifact type. This name appears in the list of options when users are selecting a new artifact to create. It also appears in the *Standard Artifact Type* column when users are viewing the artifact list.

   - `Icon`: If a new icon is added, the new icon appears in the list of options when users are selecting a new artifact to create. It also appears in the *Standard Artifact Type* column when users are viewing the artifact list. Any uploaded icon should be a PNG or JPG file that is 32x32 pixels.

   - `Prefix`: Indicates the ID prefix of the artifact type. Prefixes are unique across artifact types. All artifacts have a unique ID that begins with this prefix.

     > **Note:** The following base type prefixes are already in use within Blueprint: **AC**, **BP**, **DOC**, **DD**, **GD**, **GL**, **PF**, **PR**, **RQ**, **SB**, **UC**, **UCD**, **UM**.

   - `Tooltip`: Provides a description of the artifact type when you pause on an item with the artifact type in the artifact list.

   - `Group Label`: If specified, this artifact type appears under the specified label when users are selecting a new artifact to create. To set the group label, simply use the drop-down to select an existing group label, or type the name of a new group label into the field. If no value is specified, the artifact types appear under the *Default* label.
     In the example below, all artifact types are displayed under the *Default* label, except for the artifacts under the label called **Textual Requirement Group**:

- `Base Type`: All artifact types must have a base type. The base type determines the type of editor that is used when a user opens an artifact. The available base types are pre-configured.

- `Description`: Provides a description of the artifact type, such as the purpose or intended usage.

- `Artifact Properties`: Indicates any artifact properties that are applied to this artifact type. Place a check mark beside the artifact properties that are applicable to this artifact type. Click the **Manage Standard Properties** link to add or manage artifact properties.

4. Click **Save**.

Your standard artifact type has been successfully updated. To see any changes, close the Instance Administration Console and click **Refresh All** on the ribbon menu.

Deleting standard artifact types

> **Note**: Before you can delete a standard artifact type, you must delete all existing artifacts of that particular type.

To delete a standard artifact type:

1. Open the *Standard Artifact Types* tab.

    1. Open the Instance Administration Console.

    2. Click the **Standard Artifact Types** link.

2. Select the artifact type you want to delete.

Select an artifact type by clicking a row in the table. The artifact type details are displayed on the right side of the page.
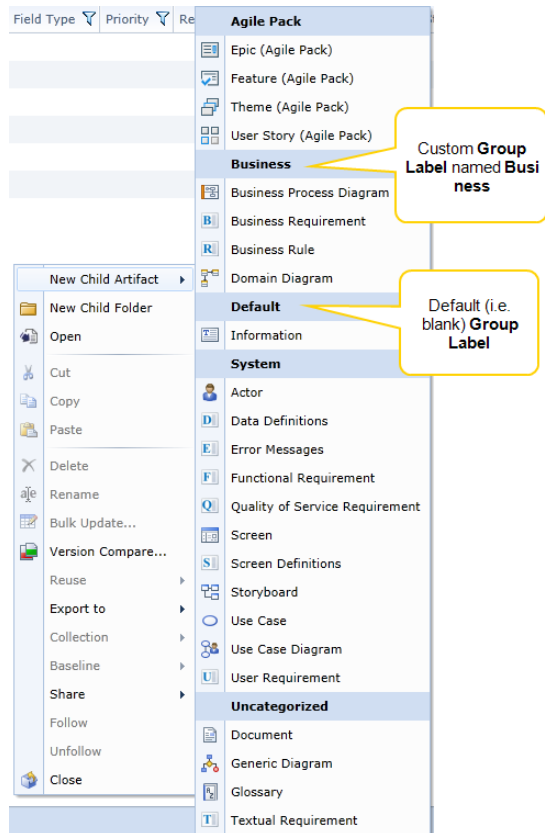
3. Click the **Delete** button on the ribbon.

   The confirmation dialog appears.

4. Click **OK** to confirm the deletion.

Your standard artifact type has been successfully deleted. To see any changes, close the Instance Administration Console and click **Refresh All** on the ribbon menu.

Creating standard artifact types

A *standard artifact type* refers to a requirement classification in Blueprint. Standard artifact types are associated with all projects in the instance.

You can create standard artifact types in order to meet the needs of your project.

> **Note:** Only Instance Administrators with the applicable privileges can create standard artifact types.

To create a new standard artifact type:

1. Open the *Standard Artifact Types* tab.

   1. Open the Instance Administration Console.

   2. Click the **Standard Artifact Types** link.

2. Click the **New** button on the ribbon.

   The new artifact type is created and the details of the artifact type are displayed in the *Standard Artifact Type Details* panel on the right side of the page.

3. Specify the artifact type details.

   - `Name`: Indicates the name of the artifact type. This name appears in the list of options when users are selecting a new artifact to create. It also appears in the *Standard Artifact Type* column when users are viewing the artifact list.

   - `Icon`: If a new icon is added, the new icon appears in the list of options when users are selecting a new artifact to create. It also appears in the *Standard Artifact Type* column when users are viewing the artifact list. Any uploaded icon should be a PNG or JPG file that is 32x32 pixels.

   - `Prefix`: Indicates the ID prefix of the artifact type. Prefixes are unique across artifact types. All artifacts have a unique ID that begins with this prefix.

     > **Note:** The following base type prefixes are already in use within Blueprint: **AC**, **BP**, **DOC**, **DD**, **GD**, **GL**, **PF**, **PR**, **RQ**, **SB**, **UC**, **UCD**, **UM**.

   - `Tooltip`: Provides a description of the artifact type when you pause on an item with the artifact type in the artifact list.

   - `Group Label`: If specified, this artifact type appears under the specified label when users are selecting a new artifact to create. To set the group label, simply use the drop-down to select an existing group label, or type the name of a new group label into the field. If no value is specified, the artifact types appear under the *Default* label.

     In the example below, all artifact types are displayed under the *Default* label, except for the artifacts

under the label called **Textual Requirement Group**:

- `Base Type`: All artifact types must have a base type. The base type determines the type of editor that is used when a user opens an artifact. The available base types are pre-configured.

- `Description`: Provides a description of the artifact type, such as the purpose or intended usage.

- `Artifact Properties`: Indicates any artifact properties that are applied to this artifact type. Place a check mark beside the artifact properties that are applicable to this artifact type. Click the **Manage Standard Properties** link to add or manage artifact properties.

4. Click **Save**.

Your standard artifact type has been successfully created. To see any changes, close the Instance Administration Console and click **Refresh All** on the ribbon menu.

About standard properties

A *standard property* refers to the descriptive data that is associated with an artifact or sub-artifact. Standard properties are configured by instance administrators with the applicable privileges and apply to all projects within an instance.

Standard properties allow you to create a single property and apply it to all projects in the instance, instead of creating the property manually in each project. By creating the property once at the instance level, you can save time and feel confident that the property is consistent across all projects.

> **Example**

> Mohammed, a business analyst, has many projects in his Blueprint instance. He wants to add a property to all of his projects in order to estimate the effort of his requirements. Instead of creating the property manually in each project, he creates a standard property called **Cost Estimation** at the instance level. The standard property is automatically applied globally to all projects, saving him time and effort.

Standard properties can be created and edited by instance administrators with the applicable privileges. Any applied changes appear uniformly across projects.

> **Tip:** Leveraging standard properties can also improve the licensed reporting feature in Blueprint Analytics. By establishing standard properties, data across projects can be consolidated easier, making filtering and modeling data more efficient.

System, standard and custom properties

Standard properties are different from default system properties that apply to all artifacts and sub-artifacts (such as **Name** and **Artifact ID**). Standard properties are also different from custom properties, which only apply to specified projects. Project Administrators cannot edit standard properties but they can assign standard properties to custom artifact types.

> **Tip:** The option to replace a custom property with a standard property is available to project administrators with the applicable privileges. Replacing a custom property with the standard effectively applies your property to all projects in the instance.

Instance administrators with the applicable privileges can create and edit standard properties and apply them to one or more artifact types. Standard properties are defined and managed on the *Standard Properties* tab in the *Instance Administration Console*. The *Standard Properties* tab looks like this:

The left side of the page provides you with a table containing the following columns of information about each custom property:

- *Name*: Indicates the name of the standard property.
- *Property Type*: The standard property can be one of the following types: *text*, *number*, *date*, *choice*, or *user*.

After you select a standard property from the table, the standard property details are displayed on the right side of the page.

Modifying standard properties

> **Warning**: Modifying a standard property can result in lost data. For example, if you change the property type, the current data is lost. Additionally, if you remove the standard property association with one or more artifact types, the standard property data is lost for those types of artifacts.

To modify an existing standard property:

1. Open the *Standard Properties* tab.

    1. Open the *Instance Administration Console*.
    2. Click the **Standard Properties** link. The *Standard Properties* tab is displayed.

2. Select the standard property you want to modify.

    Select a property by clicking a row in the table. The property details are displayed on the right side of the page.

3. Update the standard property details.

    - `Name`: Indicates the name of the standard property.
    - `Type`: The standard property can be one of the following types: *text*, *number*, *date*, *choice* or *user*.
    - `Settings`: The settings options are different depending on the selected `Type`. Here are the associated settings for each `Type`:
        - Text:
            - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
            - `Rich Text`: Defines whether or not the field supports rich text.
            - `Multi Line`: Defines whether or not the field supports multi lines of text.

- `Has Default Value`: Defines whether or not the property has a default value. If enabled, specify the default value into the space below.
  - Number
    - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
    - `Validated`: Defines whether or not the value specified for this property is validated.
    - `Number of decimal places`: Defines the number of decimal places to save.
    - `Max Value`: Defines the maximum acceptable number. This option is only applicable if the `Validated` option is enabled.
    - `Min Value`: Defines the minimum acceptable number. This option is only applicable if the `Validated` option is enabled.
    - `Has Default Value`: Defines whether or not the property has a default value. If enabled, specify the default value into the space below.
  - Date
    - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
    - `Validated`: Defines whether or not the value specified for this property is validated.
    - `Max Value`: Defines the latest acceptable date. This option is only applicable if the `Validated` option is enabled.
    - `Min Value`: Defines the earliest acceptable date. This option is only applicable if the `Validated` option is enabled.
    - `Has Default Value`: Defines whether or not the property has a default value. If enabled, specify the default value into the space below.
  - Choice
    - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
    - `Allow Custom Value`: Defines whether or not users can specify a custom value for this property.
    - `Allow Multiple Choices`: Defines whether or not users can select more than one choice for this property.
    - `Set Valid Values`: Click this button to add, delete and reorder the valid choices for this property.
  - User
    - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
    - `Has Default Value`: Defines whether or not the property has a default value. If enabled, specify the default value into the space below.
- `Applies To Artifact Types`: Place a check mark beside the standard artifact types that should contain this standard property. Click the **Manage Artifact Types** link to add or manage standard artifact types.

4. Click **Save**.

Your standard property has been successfully updated. To see any changes, close the Instance Administration Console and click **Refresh All** on the ribbon menu.

Deleting standard properties

> **Warning**: Deleting a standard property results in data loss across the instance. The standard property data is lost for all artifacts if the standard property is deleted.

To delete a standard property:

1. Open the *Standard Properties* tab.

    1. Open the *Instance Administration Console*.
    2. Click the **Standard Properties** link. The *Standard Properties* tab is displayed.

2. Click the **Delete** button on the ribbon.

    The confirmation dialog appears.

3. Click **OK** to confirm the deletion.

The standard property has been successfully deleted. To see any changes, close the Instance Administration Console and click **Refresh All** on the ribbon menu.

Creating standard properties

A *standard property* refers to the descriptive data that is associated with an artifact or sub-artifact. Standard properties are configured by instance administrators with the applicable privileges and apply to all projects within an instance.

You can create standard properties in order to make project maintenance and project editing more efficient.

To create a new standard property:

1. Open the *Standard Properties* tab.

    1. Open the *Instance Administration Console*.
    2. Click the **Standard Properties** link. The *Standard Properties* tab is displayed.

2. Click the **New** button on the ribbon.

    The new property is created and the details of the property are displayed in the *Property Details* panel on the right side of the page.

3. Specify the standard property details.

    - `Name`: Indicates the name of the standard property.
    - `Type`: The standard property can be one of the following types: *text*, *number*, *date*, *choice* or *user*.
    - `Settings`: The settings options are different depending on the selected `Type`. Here are the associated settings for each `Type`:
        - Text:
            - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
            - `Rich Text`: Defines whether or not the field supports rich text.
            - `Multi Line`: Defines whether or not the field supports multi lines of text.

- `Has Default Value`: Defines whether or not the property has a default value. If enabled, specify the default value into the space below.
  - Number
    - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
    - `Validated`: Defines whether or not the value specified for this property is validated.
    - `Number of decimal places`: Defines the number of decimal places to save.
    - `Max Value`: Defines the maximum acceptable number. This option is only applicable if the `Validated` option is enabled.
    - `Min Value`: Defines the minimum acceptable number. This option is only applicable if the `Validated` option is enabled.
    - `Has Default Value`: Defines whether or not the property has a default value. If enabled, specify the default value into the space below.
  - Date
    - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
    - `Validated`: Defines whether or not the value specified for this property is validated.
    - `Max Value`: Defines the latest acceptable date. This option is only applicable if the `Validated` option is enabled.
    - `Min Value`: Defines the earliest acceptable date. This option is only applicable if the `Validated` option is enabled.
    - `Has Default Value`: Defines whether or not the property has a default value. If enabled, specify the default value into the space below.
  - Choice
    - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
    - `Allow Custom Value`: Defines whether or not users can specify a custom value for this property.
    - `Allow Multiple Choices`: Defines whether or not users can select more than one choice for this property.
    - `Set Valid Values`: Click this button to add, delete and reorder the valid choices for this property.
  - User
    - `Required`: Defines whether or not the property is required. If the property is required, artifacts cannot be saved unless a value for this property is specified.
    - `Has Default Value`: Defines whether or not the property has a default value. If enabled, specify the default value into the space below.
- `Applies To Artifact Types`: Place a check mark beside the standard artifact types that should contain this standard property. Click the **Manage Artifact Types** link to add or manage standard artifact types.

4. Click **Save**.

> **Note:** If the *Property Conflict Resolution* dialog appears, there is a conflict you must resolve in order to proceed. For more information about resolving the conflict, see Resolving a property conflict.

Your new standard property appears in the standard properties list.

Resolving a property conflict

Overview

A property conflict can arise while creating a standard property or importing a project for a variety of reasons outlined in the following section. The resolution of a property conflict is necessary to proceed with the standard property creation or with the project import. The recommended resolution is selected by default in the *Property Conflict Resolution* dialog. An alternative resolution may be selected only when necessary and at the discretion of the instance administrator.

TYPES OF CONFLICT

The following types of conflict exist:

- **Name**

  Another existing property or artifact type has the same name as the standard property or artifact type you are attempting to create or import. Because no other conflict is detected, you may want to merge the two properties or artifact types by selecting the **Replace with Standard** option. Optionally, you can select the **Rename** option.

- **Incompatible Type**

  The standard property or artifact type you are attempting to create or import has:

  - The same name as the existing selected property or artifact type
  - A different property type from the selected property

- **Incompatible settings**

  The standard property you are attempting to create or import has:

  - The same name as the selected existing property
  - Different settings from the selected property

RESOLUTION OPTIONS

At least one of the following options is available to resolve the conflict(s):

- **Replace with Standard**

  > **Caution:** Replacing a custom property with the standard is not reversible and could result in data loss. If possible, make backups of your data before performing this operation.

You can standardize the selected custom property or artifact type. The following configurations are preserved when **Replace with Standard** is selected:

- Artifact type associations from your custom property

> **Caution:** Any existing standard property association will be lost.

- Settings from your standard property

- **Rename**

 If you rename the selected custom property, *(Non-standard)* will be attached to the name in order to differentiate it from your standard property.

To resolve a property conflict:

1. Select a conflict in the row and then select a resolution from the *Resolution* column of the *Property Conflict Resolution* dialog.

> **Note:** The recommended resolution is selected by default in the *Property Conflict Resolution* dialog. When you are unsure about a custom property, it is advisable to discuss the conflict resolution scenario with the relevant project administrator prior to selecting any resolution.

2. If necessary, repeat the first step for any other conflicts.

> **Caution:** Replacing a custom property with the standard is not reversible and could result in data loss. If possible, make backups of your data before performing this operation.

3. Click **OK**.

The conflict(s) have been resolved according to your selected action(s).

About reuse settings

Instance administrators with the applicable privileges can configure reuse settings for any standard artifact type. Instance administrators can enforce standards by controlling what information can be edited in reused artifacts. By controlling what information is read only, you can ensure only users with the correct privileges are able to change this information. Additionally, instance administrators can control whether updates to the reused artifact trigger the **suspect** icon 🛡, alerting users that the reused artifact is not aligned with the original artifact (sensitivity). *Read Only* and *Sensitive* settings are configurable in each property and can be enabled for relationships, attachments, document references and subartifacts as well.

Reuse settings be configured using the *Reuse Settings* tab in the *Instance Administration Console*, which looks like this:

The left side of the page provides you with a table containing the following columns of information about each artifact type:

- *Standard Artifact Type Name*: Indicates the name of the standard artifact type. This name appears in the list of options when users are selecting a new artifact to create. It also appears in the *Standard Artifact Type* column when users are viewing the artifact list.

- *Prefix*: Indicates the ID prefix of the standard artifact type. Prefixes are unique across artifact types. All artifacts have a unique ID that begins with this prefix.

- *Base Type*: All artifact types must have a base type. The base type determines the type of editor that is used when a user opens an artifact. The available base types are pre-configured.

After you select a standard artifact type from the table, the reuse settings you can enable are displayed on the right side of the page. Reuse settings you can enable for properties and additional data include:

- **Read Only**

  When selected, the property or data cannot be edited in any reused artifact of the same artifact type.

- **Sensitive**

  When selected, any update to the property or data triggers the **suspect** icon ⬤.

  > **Note:** By default, the **Sensitive** setting is enabled for every property, relationship, attachment, document reference and subartifact.

- **Allow override by user**

  When selected, users have the option of overriding any read only settings you have selected for the standard artifact type. When a user reuses the artifact, the *Reuse* dialog presents the option to override, as

follows:
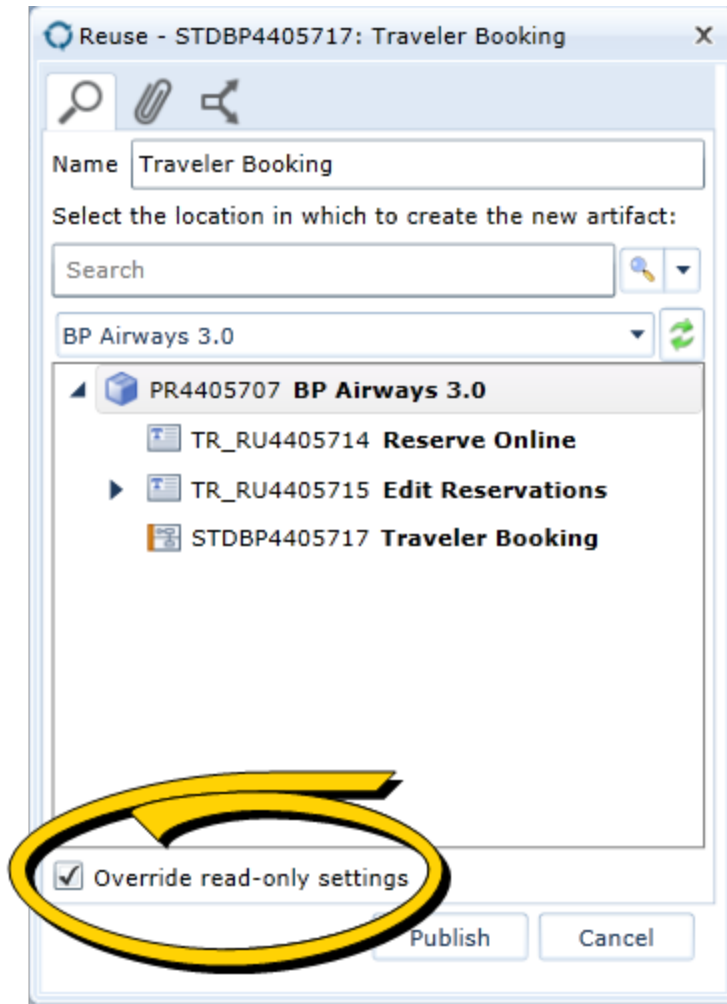
Configuring reuse settings

To configure reuse settings for a standard artifact type:

1. Open the *Reuse Settings* tab.

    1. Open the Instance Administration Console.

    2. Click the **Reuse Settings** link.

2. Select the standard artifact type you want to modify.

    Select a standard artifact type by clicking a row in the table. The reuse settings for the standard artifact type are displayed on the right side of the page.

3. Configure the reuse settings.

    - `Property Settings`: Lists the reuse settings for properties associated with the standard artifact type. When the **Read Only** check box is selected, the property cannot be edited in reused artifacts of the standard artifact type. When the **Sensitive** check box is selected, a **suspect** icon ⬤ appears

when that property is changed either in the original artifact or in the reused artifact.

- `Additional Settings`: Lists the reuse settings for relationships, attachments and document references associated with the standard artifact type. When the **Read Only** check box is selected, the data cannot be modified in reused artifacts of the standard artifact type. When the **Sensitive** check box is selected, a **suspect** icon ⬢ appears when the reused artifact is not aligned with the original artifact.

4. Click **Save**.

Your reuse settings have been successfully updated. To see your changes, close the Instance Administration Console and click **Refresh All** on the ribbon menu.

Managing projects

Creating projects

> **Important**: Instance administrators are the only users that can create projects.

Project Creation Methods

If you are an instance administrator, you can create projects using the following three methods:

- [Create empty project](#)
- [Create project from template](#)

> **Note**: Any project can be used as a template to create a new project.

- [Import a project](#)

The following table outlines the data that is included in the project, depending on the method you use to create the project:

	Empty Project	Project from Template	Imported Project
Artifact Types	Default Types Only	Yes	Yes
Custom Properties	None	Yes	Yes
Project Group Definitions (not the group members)	None	Yes	Yes
Project Roles	None	Yes	Yes
Project Role Assignments	None	Yes	No
Office Document Templates	None	Yes	Yes
ALM Targets	None	Yes	Yes
ALM Security	None	Yes	No
Folders	None	Yes	Yes

	Empty Project	Project from Template	Imported Project
Artifacts (including description and all other property values), excluding Baseline and Review Artifacts	None	Yes	Yes
Baseline and Review Artifacts	None	No	No
Artifact Traces (within the project)	None	Yes	Yes
Artifact Traces (cross-project)	None	Yes	No
Artifact Comments	None	Yes	Yes
Artifact File Attachments	None	Yes	Yes
Artifact History	None	No	No
Artifact List Saved Views	None	Yes	Yes
Reuse Relationships (within project)	None	Yes	Yes
Reuse Relationships (cross-project)	None	Yes	No
Collections	None	No	No

In summary, anything that references instance data (example: users, project role assignments, cross project data) is not exported, and therefore will not be preserved after importing the project. The artifact history is also not preserved.

Creating an empty project

An empty project contains no pre-defined groups, project roles, project role assignments, or custom properties.

> **Tip**: You can also create a project based on a template, or by importing a project. Read more about the available project creation methods.

To create an empty project:

1. Open the *Instance Administration Console*.
2. Click the **Projects** button on the ribbon.
3. Right-click the folder where you want the new project to be created, and select **New Project**.
4. Specify the project information:

   - **Name**: Defines the name of the new project.
   - **Description**: Provides a description of the project.
   - **Location**: Indicate the location of the new project.
   - **Select Source**: Select the **Empty Project** option.

5. Click **Save**.

Creating a project from a template

You can save time and improve project consistency by creating a project from a template. When you create a project from a template, the project data is preserved, such as the project groups, project roles, project role

assignments, custom properties, and so on.

In other words, you can use any existing project as a template for a new project. Therefore, it is best to create a project to use as a template, and then create new projects based on that template whenever necessary.

> **Example**
>
> Poornima, a very busy business analyst, creates new projects on a regular basis for various business units. In order to save time and ensure consistency, she decides to take advantage of the many benefits offered by templates. To start, Poornima creates a Templates folder and creates a new empty project inside the Templates folder. Poornima opens the new project and configures various roles and groups in the project, based on the standard needs across the various business units.
>
> Now, whenever a new project is required, Poornima creates a new project using project in the Templates folder as a template. In the future, Poornima may begin maintaining separate project templates for each business unit so she can easily create a customized project that meets the specific needs of each business unit.

You may be interested in other project creation methods.

To create a project from a template:

1. Open the *Instance Administration Console*.

2. Click the **Projects** button on the ribbon.

3. Right-click the folder where you want the new project to be created, and select **New Project**.

   The *New Project* dialog appears.

4. Specify the project information:

   - **Name**: Defines the name of the new project.
   - **Description**: Provides a description of the project.
   - **Location**: Indicate the location of the new project.

5. Under **Select Source**, select the **Project Template** option.

6. Click the ⋯ button.

   The *Select Project* dialog appears.

7. Select the project that you want to use as a template and then click **OK**.

8. Click **OK** on the *New Project* dialog to create the new project.

Importing a project

You can import any project that has been exported using Blueprint, or the Blueprint migration tool.

> **Important**: When you import a project, a new Blueprint project is created. You cannot import a project into an existing project (that is, merge the projects).

To import a project:

1. Open the *Instance Administration Console*.

2. Click the **Projects** button on the ribbon.

3. Select the folder where you want to import the project and click **Import Project**.

   The **Import Project** option is available by:

   - clicking **More Actions** > **Import Project** on the ribbon
   - right-clicking the folder and selecting **Import Project**.

   After you click Import Project, the *Import Project* dialog appears.

4. Click the ⋯ button and select the Blueprint project. All Blueprint exported projects are in `.zip` file format.

5. Click **Next**.

6. Specify a name for the project.

7. Click **Import**.

   > **Note**: Please be patient. The import process can take an extended period of time, depending on the size of your project. If there is a problem with the import, an error message is displayed.

   When the import is complete, the project appears in the tree on the leftmost side and the project details appear on the rightmost side.

Importing a Blueprint sample project

Importing a Blueprint sample project is a good way to become familiar with Blueprint and various customization capabilities that it offers. In addition to importing a sample project, you can also import Blueprint projects that have been exported from Blueprint or migrated from RC2010.

To import a sample Blueprint project:

1. Download the *sample project template* from the Blueprint 6.3 web page.

   > **Note**: Blueprint projects (including sample projects) are stored in a `.zip` file.

2. Open the *Instance Administration Console*.

3. Click the **Projects** button on the ribbon.

4. Select the folder where you want to import the project and click **Import Project**.

   The **Import Project** option is available by:

   - clicking **More Actions** > **Import Project** on the ribbon
   - right-clicking the folder and selecting **Import Project**.

   After you click Import Project, the *Import Project* dialog appears.

5. Click the ⋯ button and select the Blueprint project. All Blueprint exported projects are in `.zip` file format.

6. Click **Next**.

7. Specify a name for the project.

8. Click **Import**.

> **Note**: Please be patient. The import process can take an extended period of time, depending on the size of your project. If there is a problem with the import, an error message is displayed.

When the import is complete, the project appears in the tree on the leftmost side and the project details appear on the rightmost side.

Exporting projects

Exporting a project allows you to import the project back into Blueprint

### To export a project:

1. Open the *Instance Administration Console*.
2. Click the **Projects** button on the ribbon.
3. Select the project you want to export.

   > **Tip**: Try using the search field to find projects faster.

4. Click **Export Project**.

   The Export Project option is available by:

   - clicking **More Actions** > **Export Project** on the ribbon
   - right-clicking the project that you want to export and selecting **Export Project**.

   After you have click the **Export Project** button, the *Save As* dialog appears.

5. Choose the location and filename for the exported project and then click **Save**.

   A progress bar appears while the project is exporting. When the project is complete, the Export Complete dialog appears.

6. Click OK.

After you have exported a project, you can import the project to the same instance or a difference instance. Learn more about what data is preserved when you import a project.

Managing users

> **Important**: You must have the applicable Instance Administrator privileges to manage users in Blueprint. After users are added to Blueprint, Project Administrators with the correct privileges can grant access to projects by assigning users to project roles.

User Sources

With the correct Instance Administrator privileges in Blueprint, you can manually add users directly to the Blueprint database, or you can add users from your Active Directory. Blueprint supports adding users from the following sources:

- Database:

  A *Database User* is a user that is added directly to the Blueprint database. All of the user details can be modified in Blueprint. Database users must choose **Blueprint Authentication** to login to Blueprint.

- Windows:

  A *Windows User* is a user that is created using information from your Active Directory. You cannot change Windows User details in Blueprint, such as the name and password of the user. These details must be changed in the Active Directory. Windows users must choose **Windows Authentication** to login to Blueprint.

  When you create a new Windows user, the *Source* column is set to `Windows` on the *Users* tab in the *Instance Administration Console*.

User Types

There are various types of users in Blueprint:

- An *Instance Administrator* is a user that has been assigned specific role privileges at the topmost administration level (that is, the instance). Instance Administrator roles are customizable and can vary in their privileges.

- A *Project Administrator* is a user that has been assigned to a project role that contains *Project Administrator* privileges. Project Administrators are provisioned for individual projects, meaning the Project Administrator may not have Project Administrator privileges for another project.

- A *registered user* is licensed to access Blueprint and can log on to Blueprint if an Instance Administrator has enabled the ability. The level of access a registered user has varies depending on the projects that user is granted access to and the specific role they are given on each project.

- A *guest user* is someone who has been [mentioned by email address in a Blueprint discussion](). A guest user does not have a Blueprint license and cannot log on to Blueprint. A guest user can only view a snapshot of the artifact that is attached to the discussion via email. When the guest user sends an email reply, the reply is generated into a comment response in the Blueprint discussion.

- A *blocked user* is someone that has been prohibited from using the system by an Instance Administrator. A blocked user cannot receive or reply to email-integrated discussions. Only an Instance Administrator may block or unblock email addresses in the Blueprint system.

> **Note**: There is no limit to the number of instance administrators, project administrators and regular users that can exist in Blueprint.

Sorting and filtering the user list

Blueprint allows you to sort and filter the user list so you can find users and information more quickly. Filtering allows you to reduce the number of users that are displayed based on specific criteria.

You can narrow down your user list by clicking the filter icon next to any of the column headers. The filter dialog box allows you to select the criteria you want to see in the list.

After you have set a filter, you can restore your list to view all users by clicking the yellow filter icon ⧩ and then clicking the **Clear Filter** button.

Each time you click the column header, Blueprint toggles between ascending sorting, descending sorting, and no sorting. An arrow is displayed in the column header if ascending or descending sort order is activated:

> **Tip:** You can export the user list to a `.CSV` file by clicking the **Export to CSV** button at the top of the user list.

Adding database users

A *Database User* is a user that is added directly to the Blueprint database. All of the user details can be modified in Blueprint. Database users must choose **Blueprint Authentication** to login to Blueprint.

To add a database user to Blueprint:

1. Open the *Instance Administration Console*.
2. Click **Manage Users And Groups** > **Registered Users** on the ribbon (*Instance Admin* tab, *Instance* group).
3. Click **New** > **New Database User** on the ribbon (*Instance Admin* tab, *Manage Items* group).
4. Enter the user information on the right side of the window.
5. Click **Save**.

DATA ELEMENT	REQUIRED?	Description
Login Name	Yes	Defines the login name of the user. This field is alphanumeric and must be between 4 and 255 characters.
Source	N/A	Defines the user source. This value is automatically set to either Database or Windows, depending on the type of user account.
Enable Login	N/A	Defines whether or not the user can login to the system. By default, this option is selected, permitting login access to the user.
Instance Administrator Role	N/A	Defines whether or not the user has Instance Administrator role privileges. By default, no role is assigned.
Allow fallback from federated authentication	N/A	Defines whether or not the user is allowed to fallback from federated authentication. This option is only available if federated authentication is enabled.
Picture	No	To add or change a picture, click the **Edit** button to choose an image file. To remove a picture, click the **Remove** button.
Display Name	Yes	Defines the display name of the user.
First Name	Yes	Defines user's first name.
Last Name	Yes	Defines the user's last name.
Password / Confirm Password	Depends	Defines the password of the user. The password must be between 6 and 14 characters. The password field is no longer visible after the user is saved. **Note**: If federated authentication is enabled, a password is not required when fallback from federated authentication is not enabled.
Email	No	Defines the e-mail address of the user.
Title	No	Defines the job title of the user.
Department	No	Defines the department to which the user belongs.
Group Membership	No	Defines the groups to which the user is a member. If you want to add a user to an existing group, click Add to view a list of available groups.

Adding Windows users

A *Windows User* is a user that is created using information from your Active Directory. You cannot change Windows User details in Blueprint, such as the name and password of the user. These details must be changed in the Active Directory. Windows users must choose **Windows Authentication** to login to Blueprint.

After you add a new user to Blueprint, you must assign the user to a license group. If the user is not added to a license group, the user is limited to view privileges in Blueprint.

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

To add Windows users to Blueprint:

1. Open the *Instance Administration Console*.
2. Click **Manage Users And Groups** > **Registered Users** on the ribbon (*Instance Admin* tab, *Instance* group).
3. Click **New** > **New Windows User** on the ribbon (*Instance Admin* tab, *Manage Items* group).
4. Click the **Find** button to display all Active Directory users.

   If multiple active directory servers are configured, you can change the **Connection** option to select a different active directory server. The **Connection** option only appears if multiple active directory servers are defined.

5. Select the users you want to add, or type `Ctrl-a` to select all users.
6. Click **OK**.

Assigning an Instance Administrator role to a user

An *Instance Administrator* is a user that has been assigned specific role privileges at the topmost administration level (that is, the instance). Instance Administrator roles are customizable and can vary in their privileges.

Instance Administrator privileges are assigned on a user basis. You cannot assign *Instance Administrator* privileges to a group.

To assign instance administrator privileges to a user:

1. Open the *Instance Administration Console*.
2. Click **Manage Users And Groups** > **Registered Users** on the ribbon (*Instance Admin* tab, *Instance* group).
3. Click the user to which you want to grant *instance administrator* privileges.
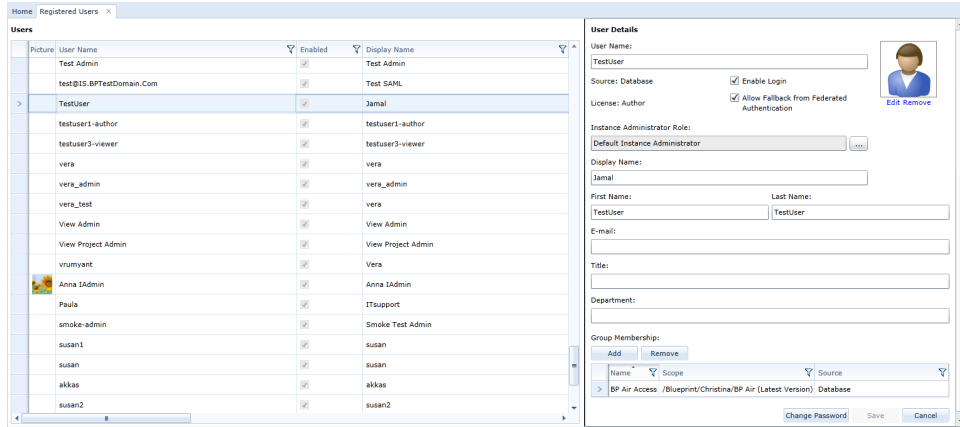
4. Click the **…** button next to the *Instance Administrator Role* field.
   The *Instance Administrator Assignment* dialog box appears.

5. Select the role you want to assign from the **Instance Administrator Role** drop-down box.
6. Click **OK**.
7. Click **Save**.

The role has successfully been applied to the user account.

Assigning a Project Administrator role to a user

A *Project Administrator* is a user that has been assigned to a project role that contains *Project Administrator* privileges. Project Administrators are provisioned for individual projects, meaning the Project Administrator may not have Project Administrator privileges for another project.

> **Tip**: Because project administrator privileges are granted using project roles, you can assign the project admin privilege to an individual user or to an entire group. You may want to consider creating a **Project Administrators** group and then assign that group to a new project role called **Project Administrator Role**. If you use this strategy, you can simply add a user to the **Project Administrators** group to grant project administrator privileges to that user.

Project administrators with the correct privileges can grant project role privileges to any user or group, but only within the project(s) to which the user has *project administrator* privileges.

> **Note:** Project administrators can only assign custom project administrator roles if the roles have already been created by an instance administrator.

To grant project administrator privileges to a user or group:

1. Create the new user or new group if it does not already exist (Instance Administration Console).

   > **Note**: You must have the correct instance administrative privileges to create a new user.

2. Create the project administrator role if it does not already exist (Instance Administration Console).
3. Create a project role (Project Administration Console). This project role must be created in each Blueprint project if you want users to have project administrator privileges.

Select the role from the **Project Administrator Role** menu containing the privileges you want to give the user or group.

4. Create a new project role assignment to assign the user (or group) to the project role that contains the project administrator privilege (Project Administration Console).

Modifying a Windows user

After you have added Windows users to Blueprint, you can modify various user data and options.

To modify a Windows user in Blueprint:

1. Open the *Instance Administration Console*.
2. Click **Manage Users And Groups** > **Registered Users** on the ribbon (*Instance Admin* tab, *Instance* group).
3. Click the user you want to modify.
4. You can modify the following user data and options:

DATA ELEMENT	REQUIRED?	Description
Enable Login	N/A	Defines whether or not the user can login to the system. By default, this option is selected, permitting login access to the user.
Instance Administrator Role	N/A	Defines whether or not the user has Instance Administrator role privileges. By default, no role is assigned.
Allow fallback from federated authentication	N/A	Defines whether or not the user is allowed to fallback from federated authentication. This option is only available if federated authentication is enabled.
Picture	No	To add or change a picture, click the **Edit** button to choose an image file. To remove a picture, click the **Remove** button.
Email	No	Defines the e-mail address of the user.
Group Membership	No	Defines the groups to which the user is a member. If you want to add a user to an existing group, click **Add** to view a list of available groups.

5. Click **Save**.

Managing instance-level groups

Each group consists of a name, e-mail address, scope, and source. Groups can be created at the instance level and be given a project-level scope. If a group scope is not defined, the group can be viewed and used (but not

modified) at the project level.

Instance-level groups can be designated as a license group. License groups allow you to control the type of license that a user holds after logging into the system. If a user does not belong to a Author or Collaborator license group, the user is automatically granted a View license. Read more about [managing licenses](#).

Instance-level groups are managed using the *Groups* tab in the *Instance Administration Console*. When you open the *Groups* tab, the groups are displayed in the leftmost panel, and the group details are displayed in the rightmost panel.

> **Note**: The *Groups* tab in the *Instance Administration Console* displays all instance-level groups and all project-level groups in the instance. Instance administrators can create, modify, and delete all instance-level and project-level groups.

The *Groups* tab looks like this:
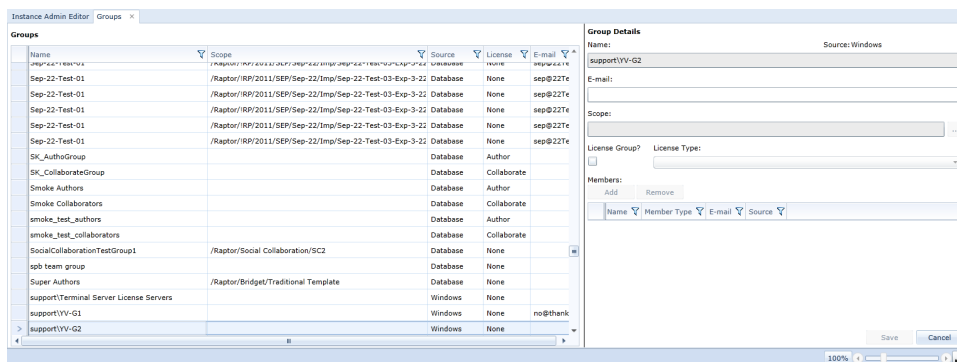
Understanding the Groups Tab

The Groups tab is accessible from both the Instance Administration Console and the Project Administration Console, but your ability to perform certain operations differs slightly depending on whether you are an instance administrator or a project administrator.

The left side of the *Groups* tab provides you with a table containing the following columns of information about each group:

- *Name*: Indicates the name of the group.
- *Email*: Indicates the group email address.
- *Scope*: Indicates the scope of the group. If a scope is defined, the group is only visible at the specified project level. If no scope is defined, the group is visible within all projects.

> **Note**: Groups that are created at the instance-level cannot be modified by project administrators, regardless of the group scope.

- *Source*: Indicates the source of the group. The value in this column can be either *Database* or *Windows*. Blueprint only allows project administrators with the applicable privileges to manage *Database* groups. *Windows* groups are derived from the Active Directory, therefore cannot be managed by the project administrator.

After you select a group from the table, the group details are displayed on the right side of the page, including the list of group members. The group members are displayed in a table with the following columns of information about each member:

- *Name*: Indicates the name of the group member.
- *Member Type*: Indicates whether the group member is a user or a group.

> **Tip**: You can add groups to other groups.

- *Email*: Indicates the email address of the group member.
- *Source*: Indicates whether the group member source is *Windows* (Active Directory) or the Blueprint *Database*.

> **Note**: Both *Windows* and *Database* member sources can be added to a *Database* group.

- *Scope*: Indicates the scope of the group.
- *License Type*: Indicates whether the license is Author, Collaborator or View.

Creating instance-level groups

Instance-level groups can consist of numerous users and/or groups. Groups make it easier to provide access to projects because you can assign a role to an entire group instead of individual users.

Instance-level groups can be designated as a license group. License groups allow you to control the type of license that a user holds after logging into the system. If a user does not belong to a Author or Collaborator license group, the user is automatically granted a View license. Read more about managing licenses.

> **Note**: Instance-level groups can be used, but not modified, by project administrators.

To create an instance-level group:

1. Open the *Instance Administration Console*.
2. Click **Manage Users And Groups** > **Groups** on the ribbon (*Instance Admin* tab, *Instance* group).
3. Click **New** > **Database Group** on the ribbon (*Instance Admin* tab, *Manage Items* group).
4. Enter the group information:

   - **Name**: Defines the name of the group.
   - **Description**: Provides a description of the group.
   - **Email**: Defines the group email address.
   - **Scope**: Defines the scope of the group. For instance-level groups, it is usually best to leave this field blank so the group can be accessed by all projects in the instance. You can, however, set this field if you want to limit the scope of the instance-level group to a particular set of projects.

     > **Important**: The scope cannot be changed after you save the group.

   - **License Group?**: Select this option if you want this group to be a license group. If selected, the **Scope** must be left blank because license groups must be instance-level groups, not project-level groups.
   - **License Type**: If the **License Group** option is enabled, select the type of license for this group. There are two types of license groups that you can create, but there are three license types in total:

- **Author**: An *author* license allows users to author requirements in Blueprint, as well as perform all of the tasks that a user with a *collaborate* license can perform.
- **Collaborator**: A *collaborate* license allows users to login to Blueprint to perform tasks such as simulating use cases and participating in reviews.
- **View**: A *view* license only allows users to access a single artifact accessed by the artifact URL. The user can view the artifact properties, comments, traces, and historical information. You cannot create a *view* license group because the view license is the default license type.

5. Click the **Add** button to add members to the group.
6. Select the users or groups that you want to add. You can type `Ctrl-a` to select all users.
7. Click **OK**.
8. Click **Save**.

About managing administrator roles

Overview

Administrator roles contain privileges that are required to perform management tasks at the project level as well as at the higher instance level.

Blueprint offers default administrator roles for project administration and instance level administration as well as the ability to create custom administrator roles.

Default administrator roles

Within the *Instance Admin Editor*, Blueprint provides a Default Project Administrator role and a Default Instance Administrator role that contain all privileges to their respective areas, the *Project Administration Console* and the *Instance Administration Console*.

Blueprint also provides other default administrator roles with varying privileges, which you can modify.

Default instance administrator roles

You can modify any of the following roles that Blueprint provides for instance administration:

- Log Gathering and License Reporting

  The administrator can download all instance logs, as well as generate and download the License Report. The administrator can view but not manage the following settings: federated authentication settings, the instance-level print template, active directory integration, email settings and file size settings.

- Email, Active Directory, SAML Settings

  The administrator can download all instance logs, as well as generate and download the License Report. The administrator can also manage the following: federated authentication settings, the instance-level print template, active directory integration, email settings and file size settings.

- Manage Administrator Roles

  The administrator can manage all Instance Administrator and Project Administrator roles. This role includes the ability to create new custom roles.

- Provision Users

  The administrator can provision new users and groups as well as manage existing users and groups.

- Provision Projects

  The administrator can create new projects and modify existing projects, as well as import and export projects.

  > **Note**: An administrator with this role cannot delete projects.

- Administer All Projects

  The administrator can create new projects and modify all existing projects. The administrator has the full privileges of a default Project Administrator role.

  > **Note**: An administrator with this role cannot delete projects.

- Assign Instance Administrators

  The administrator can create and manage the list of users, as well as assign instance administrator roles to users.

Default project administrator roles

You can modify any of the following roles that Blueprint provides for project administration:

- Grant Project Access

  The administrator can create project roles and give role assignments to users. The administrator can also create and manage project-level groups.

- Manage Project Configuration

  The administrator can modify the project details, manage the default print template and Office Document templates, modify comment settings and extract the project XML file.

- ALM Target Administrator

  The administrator can manage ALM targets and security.

Creating administrator roles

Blueprint provides large and growing enterprises with the ability to create customized administrator roles. Customizing administrator roles limits administrator access to specific areas and privileges. For example, you could create an administrator role that can manage projects but cannot manage users and groups.

The benefits of creating custom administrator roles include:

- Preventing unnecessary data loss
- Meeting auditing requirements
- Isolating potential mistakes by employees

> **Example**
> Jesse, a manager, wants to give an employee (Sam, a business analyst) administrative abilities. However, Jesse wants to minimize unintended deletions and project data loss. In order to achieve his goals, Jesse creates a custom instance administrator role with project deletion privileges and

assigns it to an administrator in the IT department. Next, Jesse creates a custom instance administrator role with the ability to manage projects and, also, the ability to view users and groups; then he assigns the new custom role to Sam. By creating custom instance administrator roles and assigning them to the appropriate parties, Jesse can maintain security while also achieving his management goals.

Role privileges are generally categorized into access, view, manage, assign, delete and edit actions with access divided into users, groups, projects and instance settings.

About Instance Administrator role privileges

Overview

In the Instance Administration Console, Blueprint provides you with the ability to create custom administrative roles.

> **Note:** When you deselect an instance administrator privilege, the privilege is grayed-out and inaccessible to the user.

After creating a custom instance administration role, you must assign the role to user(s) in order for the privileges to take effect. For more information about assigning a custom instance administration role to a user, see Assigning an Instance Administrator role to a user.
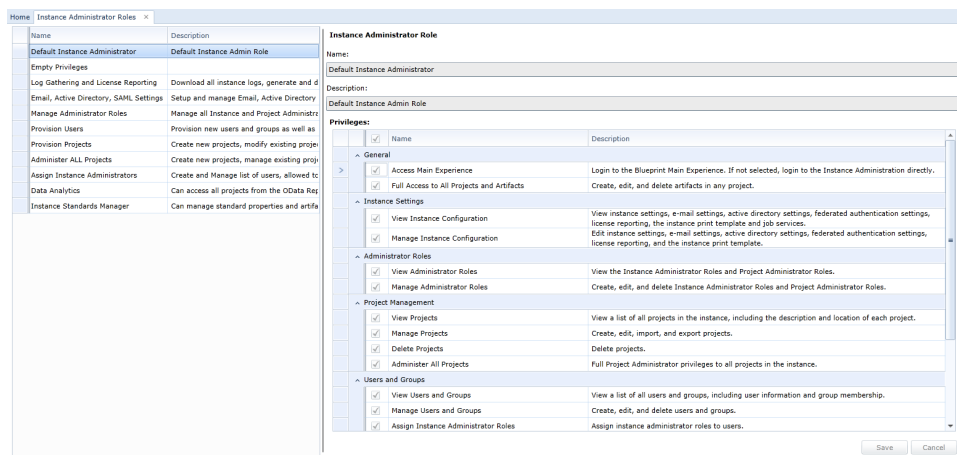
The privileges that are available for custom Instance Administrator role building are outlined in the following sections.

> **Note:** Default Instance Administrators have all of the privileges described in the sections below. For more information about default roles, see Default administrator roles.

Access Main Experience

When selected, the administrator accesses the default Blueprint experience upon logging on. The default page is the Activity Center.

When deselected, the administrator accesses the Instance Admin upon logging on.

Full Access to All Projects and Artifacts

When selected, the administrator has access to all projects and artifacts. The administrator can create, edit and delete artifacts in any project.

View Instance Configuration

When selected, the administrator can view but cannot edit the following Instance Settings:

- Rich Text Settings
- File Settings
- Logging: you can download the log
- Quick Links
- Email Settings
- Active Directory Settings
- Federated Authentication Settings
- License Reporting
- Instance Print Template
- Job Services

> **Note:** The *View Only* label appears when a user does not have access to modify the setting.

Manage Instance Configuration

When selected, the administrator can modify the following Instance Settings:

- Rich Text Settings
- File Settings
- Logging: you can download the log
- Quick Links
- Email Settings
- Active Directory Settings
- Federated Authentication Settings
- License Reporting
- Instance Print Template
- Job Services

View Administrator Roles

When selected, the administrator can view but cannot modify the following administrator roles:

- Instance Administrator roles
- Project Administrator roles

> **Note:** The *View Only* label appears when a user does not have access to modify the setting.

Manage Administrator Roles

When selected, the administrator can create, modify and delete the following administrator roles:

- Instance Administrator roles
- Project Administrator roles

View Projects

When selected, the administrator can view but cannot edit the list of projects, including the description and location of each project. The administrator can open the **Projects** experience.

Manage Projects

When selected, the administrator can create, modify, import and export projects.
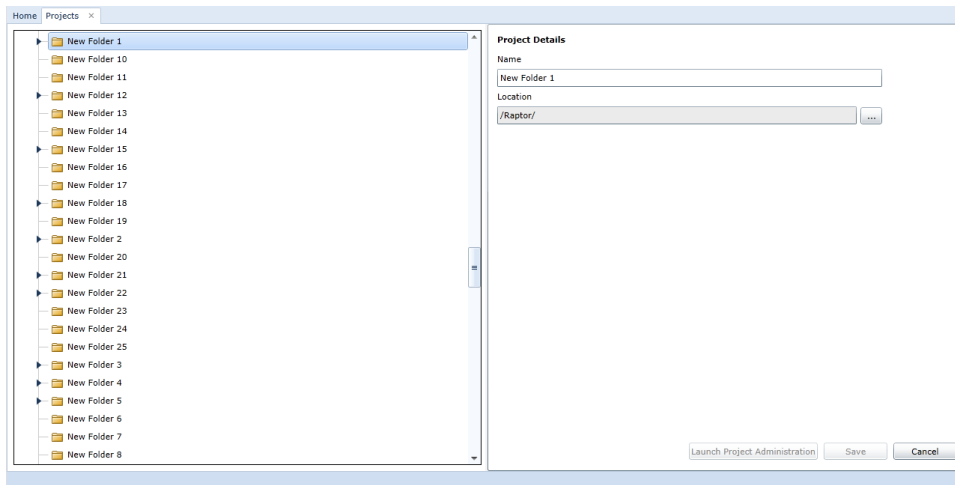
> **Note:** An administrator with the *Manage Projects* privilege cannot access the **Launch Project Administration** button within *Projects*.

Delete Projects

When selected, the administrator can delete projects and folders.

Administer All Projects

When selected, the administrator can access the Project Administration Console for all projects. The administrator can manage projects and has the full privileges of a default Project Administrator role.

View Users and Groups

When selected, the administrator can view the user list and group list, including user information and group membership. However, the administrator cannot edit the user list and group list.

> **Note:** The *View Only* label appears when a user does not have access to modify the setting.

Manage Users and Groups

When selected, the administrator can create, edit and delete users and groups. The administrator can also sync Windows users (*Actions* group, **More Actions** button).

Assign Instance Administrator Roles

When selected, the administrator can assign Instance Administrator roles to users.

Produce Blueprint Analytics Reports On All Projects

When selected, the administrator can produce Blueprint Analytics reports using data from any and all Blueprint projects.

> **Note:** Blueprint Analytics reporting requires a Blueprint Analytics license.

View Standard Properties and Standard Artifact Types

When selected, the administrator can view a list of standard properties, including standard property settings and standard artifact type assignments. The administrator can also view a list of standard artifact types, including group labels, base types and standard artifact properties. However, the administrator cannot edit the standard properties and standard artifact types.

Manage Standard Properties and Standard Artifact Types

When selected, the administrator can create, edit and delete standard properties, including standard property settings and standard artifact type assignments. The administrator can also create, edit and delete standard artifact types, including group labels, base types and standard artifact properties.

When this privilege is selected, the following privileges are also automatically granted:

- View Projects
- Manage Projects
- Administer All Projects
- View Standard Properties and Standard Artifact Types

Manage All Running Jobs

When selected, the administrator can cancel any running job.

Creating a new Instance Administrator role

To add a new Instance Administrator role:

1. Open the *Instance Administration Console*.
2. Click **Instance Administrator Roles** (*Manage Administrator Roles* group).
   The *Instance Administrator Roles* page appears.
3. Click the **New** icon (*Manage Items* group).
4. In the *Instance Administrator Role* pane, enter the name of your role in the **Name** field.

   Enter a description in the **Description** field so you know what privileges you are selecting when you assign the instance administrator role.

5. In the *Role Privileges* list, select the privileges you want the administrator role to have.
6. Click **Save**.

The new role appears in the list of Instance Administrator Roles.

To assign the new role to a user, see Assigning an Instance Administrator role to a user.

About Project Administrator role privileges

Overview

In the *Instance Administration Console*, Blueprint provides you with the ability to create custom administrative roles.

After creating a custom project administrator role, the project administrator role must be assigned to a project role and then the project role must be assigned to a user or group in order for the privileges to take effect. For more information about assigning a custom project administration role to a user, see Assigning a Project Administrator role to a user.

The privileges that are available for custom Project Administrator role building are outlined in the following sections.

> **Note:** Default Project Administrators have all of the privileges described in the sections below. For more information about default roles, see Default administrator roles.

View Groups, Project Roles and Project Role Assignments

When selected, the user can view the group list, the project role list and the project role assignment list but cannot edit the group list, the project role list or the project role assignment list.

> **Note:** The *View Only* label appears when a user does not have access to modify the setting.

Manage Groups and Project Roles

When selected, the user can modify groups, project roles and project role assignments.

View Project Configuration

When selected, the user can view but not edit the following:

- Project Details
- Project Settings
- Project Print Template
- Custom Properties (*Customize Project* section)
- Custom Artifact Types (*Customize Project* section)
- Sub-artifacts (*Customize Project* section)
- Status Values (*Customize Project* section)
- Office Document Templates

> **Note:** The *View Only* label appears when a user does not have access to modify the setting.

Manage Project Configuration

When selected, the user can edit the following:

- Project Details
- Project Settings
- Project Print Template
- Custom Properties (*Customize Project* section)
- Custom Artifact Types (*Customize Project* section)
- Sub-artifacts (*Customize Project* section)
- Status Values (*Customize Project* section)
- Office Document Templates

View ALM Integration Settings

When selected, the user can view but not edit the following:

- ALM Targets (*ALM Integration* section)
- ALM Security (*ALM Integration* section)

> **Note:** The *View Only* label appears when a user does not have access to modify the setting.

Manage ALM Integration Settings

When selected, the user can view but not edit the following:

- ALM Targets (*ALM Integration* section)
- ALM Security (*ALM Integration* section)

Creating a new Project Administrator role

To add a new Project Administrator role:

1. Open the *Instance Administration Console*.
2. Click **Project Administrator Roles** (*Manage Administrator Roles* group).
3. Click the **New** icon (*Manage Items* group).
4. In the *Project Administrator Role* pane, enter the name of your role in the **Name** field.

   Enter a description in the **Description** field so you know what privileges you are selecting when you assign the instance administrator role.

5. In the *Role Privileges* list, select the privileges you want the administrator role to have.
6. Click **Save**.

Managing licenses

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

> **Examples**
>
> If William has a *collaborator* license, he is not able to edit requirements even if a project role assignment grants him **Edit** privileges. William needs an *author* license to utilize the **Edit** privileges that are set in the project role assignment. In other words, licenses can limit/override the privileges outlined in a project role assignment to ensure that licenses are enforced.
>
> If Brenda has an *author* license, she is not able to edit requirements if a project role assignment only grants her **Read** and **Comment** privileges. In this case, the project role assignment permissions are respected and Brenda is limited to reading and commenting on artifacts even though she has an author license. In other words, the license assignment does not automatically grant access to users in the absence of a project role assignment.

There are three types of Blueprint licenses:

- **Author**: An *author* license allows users to author requirements in Blueprint, as well as perform all of the tasks that a user with a *collaborate* license can perform.
- **Collaborator**: A *collaborate* license allows users to login to Blueprint to perform tasks such as simulating use cases and participating in reviews.
- **View**: A *view* license only allows users to access a single artifact accessed by the artifact URL. The user can view the artifact properties, comments, traces, and historical information. You cannot create a *view* license group because the view license is the default license type.

Maximum capabilities by license type

> **Important**: Project role assignments are used to grant privileges to users so they can access artifacts in Blueprint. Licenses simply enforce a maximum capability level.

Capability	Author License	Collaborate License	View License
View single requirement via URL	✔	✔	✔
Browse and view requirements	✔	✔	
Create and participate in discussions	✔	✔	
Participate in requirements reviews	✔	✔	
Simulate requirements	✔	✔	
Setup and manage requirements reviews	✔		
Create and edit requirements	✔		
Create and edit projects	✔		
Generate documents and import/export	✔		
Push requirements to ALM systems	✔		

Viewing license reports

License reports provide instance administrators with the information necessary to manage licenses effectively. For example, as an instance administrator with the correct privileges, you can determine how many licenses are currently available.

There are two types of reports:

- License Status: Provides information about the current status of licenses.
- License Activity Report: Provides information about past license usage, such as the maximum concurrent usage, and detailed license transactions.

License Status Report

The license status report is automatically updated when you open the License Reporting tab in the Instance Administration Console. The License Status Report looks like this:

License	Maximum possible	Current	Available Room	Expiry Date
Author	100	1	99	None
Collaborate	100	0	100	None
View	Unlimited	0	Unlimited	None

The license status report consists of 5 columns:

- *License*: This column lists each Blueprint license type.
- *Maximum possible*: Indicates the total number of purchased licenses, by license type.
- *Current*: Indicates the total number of licenses that are currently in use, by license type.
- *Available Room*: Indicates the total number of available (unused) licenses, by license type.
- *Expiry Date*: Indicates the expiry date of the license.

License Activity Report

The license activity report always provides data using data from the past, where the number of days is configurable. For example, you can generate a report using data from the past 7 days or the past 30 days. The License Activity Report looks like this:

Report on the last [30] days. [Go]

High Water Mark

License	Count	Last Date
Author	17	08/06/2012 6:23:31 PM
Collaborate	2	11/06/2012 7:50:36 PM
View	2	21/06/2012 7:33:02 PM

License Transactions

Date	Type	Action	Username	Department	License Type	Total Authors	Total Collaborators	Total Viewers
20/06/2012 3:12:19 PM	Acquire	Login	susan4		View	10	0	1
20/06/2012 3:12:01 PM	Release	Logout	susan3		Collaborate	10	0	0
20/06/2012 3:09:33 PM	Acquire	Login	blueprint\praveendran		Author	10	1	0
20/06/2012 3:07:25 PM	Release	Logout	yv-3		Collaborate	9	1	0
20/06/2012 3:07:05 PM	Acquire	Login	blueprint\scostiuc		Author	9	2	0
20/06/2012 3:06:47 PM	Release	Logout	2admin	HR	Author	8	2	0
20/06/2012 3:06:07 PM	Acquire	Login	blueprint\pvincent		Author	9	2	0

[Download to CSV]

The License Transactions report consists of the following columns:

- *Date*: Indicates the date and time when the transaction occurred.
- *Type*: Indicates the type of transaction.
- *Action*: Indicates the action that triggered the license transaction.
- *Username*: Indicates the user that triggered the license transaction.
- *Department*: Indicates the department of the user that triggered the license transaction.
- *License Type*: Indicates the type of license involved in the transaction.
- *Total Authors*: Indicates the total number of author licenses in use when the transaction occurred.
- *Total Collaborators*: Indicates the total number of collaborator licenses in use when the transaction occurred.
- *Total Viewers*: Indicates the total number of view licenses in use when the transaction occurred.

Viewing the Blueprint license reports:

1. Open the *Instance Administration Console*.

2. Click the **License Reporting** link on the *Instance Admin Editor* tab.

   The License Status report is automatically generated. To view the License Activity Report, select the number of days you want the report to include and then click the **Go** button.

   You can download the License Transactions by clicking the **Download to CSV** button.

   > **Note**: If you are using Internet Explorer 8, you must enable the *automatic prompting for file downloads* security setting before you can download the file from Blueprint. To enable this setting, click **Tools** > **Internet Options** > **Security** > **Custom level...** > **Downloads** and then enable the **Automatic prompting for file downloads** option.

Managing instance settings

Instance administrators with the applicable privileges can configure various settings, including file size limits, artifact usage limits and *Home Page* quick links. Within the *Instance Settings*, you can also download the Blueprint log, which can be helpful if you are troubleshooting an issue (*Logging* tab).

The *Instance Settings* looks like this:

Blueprint gives you the ability to configure various aspects of the instance on the *General* tab. You can configure the following settings:

- Rich Text Properties

  Blueprint gives you the ability to set the default font family and font size of any newly created artifact within the instance. This setting applies to rich text supported fields only.

- Activity Center Settings

  Blueprint gives you the ability to set the number of past months you want to show in the activity feed by default.

- File Settings

- Excel Update Properties

  You can enable or disable the ability to update artifacts using Excel spreadsheets across the instance.

- Artifact Reuse Properties

  Blueprint gives you the ability to let users edit and update the writable properties of reused artifacts.

Configuring files

As a Blueprint instance administrator with the correct privileges, you can control various file settings in Blueprint. For example, you may want to control the maximum size of files that users can upload to Blueprint.

To configure file settings:

1. Open the *Instance Administration Console*.
2. Click **Manage > Instance Settings** on the ribbon.
3. Specify the file settings on the *General* tab:
   - **Max File Size**: Defines the maximum size of files that users can upload to Blueprint. This value controls the size of files that are uploaded to document artifacts, as well as file attachments that can be added to any type of artifact.
   - **File Size Warning Threshold**: Defines the file size at which a warning message is displayed to the user. For example, you may want to set the **Max File Size** to 10MB so large files are accepted, but then use the **File Size Warning Threshold** setting to warn users for all files greater than 3MB in size.
4. Click **Save**.

About Blueprint logging

Overview

Within the Instance Settings, Blueprint provides a log zip file that contains the following log files:

- Server log

  Provides information about debugging and errors that have occurred in order to help you with troubleshooting.
- Audit log (CSV file)

  Provides a record of changes administrators have made within the Instance Administration Console and the Project Administration Console.
- API log

  Provides activity, error and debugging information for API developers.
- Client log

  Provides information about Blueprint user activity in order to help you with troubleshooting.

For information about downloading the log zip file, see Downloading the Blueprint log zip file.

About the client log

Overview

The client log provides information about Blueprint user activity in order to help you with client troubleshooting. The log is available within the Blueprint log zip file (*Instance Administration Console*).

The log file is provided as a comma-separated file that lists each log entry on a new line. Here's an example of a single log entry:

```
Client: 21/03/2014 12:24:43, GMT-05:00, acme\wecoyote, Info,Information
message: Shared View settings are applied.
```

Here's an explanation of the data contained in each log entry, outlined from left to right:

Log Entry Data	Description	Example
Date/Time	Indicates the date and time that the item was logged.	`21/03/2014 12:24:43`
Time Zone	Indicates the time zone of the Date/Time value that was logged.	`GMT-05:00`
User	Indicates the user name of the user that triggered the log entry.	`acme\wecoyote`
Type	Indicates the type of log message. This value can be set to: ■ Info ■ Debug ■ Warn ■ Error ■ Fatal	`Info`
Action	Provides a detailed summary of the log entry. This information can be useful for the Blueprint Support team if you require assistance troubleshooting a problem.	`Information message: Shared View settings are applied.`

Tip: To download a log of an individual user's activity: using the user's account, click the *application menu* [Menu] and then click **Profile Options**. When the dialog appears, click the **Download** button under the *Client log* section.

To view the client logs:

Note: If you are using Internet Explorer 8, you must enable the *automatic prompting for file downloads* security setting before you can download the file from Blueprint. To enable this setting, click **Tools** > **Internet Options** > **Security** > **Custom level...** > **Downloads** and then enable the **Automatic prompting for file downloads** option.

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

3. Click **Logging**.

4. Click the **Download Log** button.

   The download dialog appears.

5. Click **Open**.

   The file is unzipped.

6. Open `Blueprint.Client.Log` in a text editor.

The client log file appears.

About the server log

Overview

The Blueprint server log file provides information about system usage. This information is useful for troubleshooting purposes.

The log file is provided as a comma-separated file that lists each log entry on a new line. Here's an example of a single log entry:

```
04/07/2012 11:27:12 AM,GMT-
05:00,dmnptkx5w5dvrseyxtr2pwyh,acme\wecoyote,Info,ChangeSummary - Finish -
for (int i =(path.Count -1);i>=0;i--)
```

Here's an explanation of the data contained in each log entry, outlined from left to right:

Log Entry Data	Description	Example
Date/Time	Indicates the date and time that the item was logged.	`04/07/2012 11:27:12 AM`
Time Zone	Indicates the time zone of the Date/Time value that was logged.	`GMT-05:00`
Session ID	Indicates the session ID of the user that triggered the log entry.	`dmnptkx5w5dvrseyxtr2pwyh`
User	Indicates the user name of the user that triggered the log entry.	`acme\wecoyote`
Type	Indicates the type of log message. This value can be set to: ■ Info ■ Debug ■ Warn ■ Error ■ Fatal	`Info`

Log Entry Data	Description	Example
Action	Indicates whether or not the log entry occurred due to a login or logout action. This value can be set to: • [blank] • Login • Logout	`Login`
Change Summary	Provides a detailed summary of the log entry. This information can be useful for the Blueprint Support team if you require assistance troubleshooting a problem.	`ChangeSummary - Start - for (int i =(path.Count - 1);i>=0;i--)`

About the audit log

Overview

> **Note:** Because it is geared towards maintaining security within an enterprise structure, the audit log is only accessible at the Instance Administration level.

The audit log provides a detailed record of administrative activities, helping you keep of track important operations that have taken place within the system. Whereas artifact versioning and history allows you to view the changes that have occurred in an individual artifact or project, the audit log provides an account of administrative actions that have taken place within the Instance Administration Console and the Project Administration Console. For example, audit logging can facilitate insight into a variety of administrative activities, such as granted privileges and modified instance settings.

Audit logging provides the additional benefit of helping you troubleshoot high-level issues effectively.

Provided in the main log zip file, the audit log is a CSV file that lists each log entry on a new line. Here's an example of an audit log:

> **Important:** Depending on whether the log entry category is applicable to the action that occurred, the log entry field either contains data or is blank.

Here's an explanation of the data contained in each log entry, outlined in the order the columns appear:

Log Entry Data	Description	Example
DateTime	Indicates the date that the action was logged.	`05/23/2013 14:40:45`
UserID	Indicates the ID of the user that performed the logged action.	`348`
UserName	Indicates the user name of the user that performed the logged action.	`jsmith`

Log Entry Data	Description	Example
Scope	Indicates if the action was instance-wide or specific to a project. The scope can be either: ■ Project --Or-- ■ Instance	`Project`
ProjectID	If the logged action was specific to a project, indicates the ID of the project.	`74840`
ProjectName	If the logged action was specific to a project, indicates the project name.	`OnlineBankingProject`
Area	Indicates the functional area that the action applies to. Possible areas include: ■ Users ■ Groups ■ Roles ■ Group Assignment ■ Projects ■ Project Role Assignments - User ■ Project Role Assignments - Group ■ Project Settings ■ Properties ■ Property Assignments	`Groups`
Action	Indicates the type of action that the user performed. Possible actions include: ■ Add ■ Edit ■ Delete	`Edit`
ObjectId	Indicates the ID of the object that the user acted upon.	`571`
ObjectName	Indicates the name of the object that the action was applied to. An object can be a wide variety of things, including (but not limited to): projects, artifact types, custom properties, users, roles, groups, ALM integrations, document generation templates.	`Collaborator`
Attribute	Indicates the attribute that the action applies to.	`Email`
NewValue	Indicates the new value that the attribute has been set to.	`newemail@address.com`
OldValue	Indicates the previous value of the attribute.	`oldemail@address.com`
Details	Depending on the object type that has been added, removed or edited, provides any additional details.	`TYPE=Database EMAIL= SCOPE=/MainProject ISLICENSED=False`

blueprint®

About the API log

Overview

The API log provides a detailed record of API requests and responses, helping you keep of track important actions that have been performed on Blueprint data.

The log is provided as a comma-separate file that lists each new entry on a new line. Here's an example of a single log entry:

```
Error, 02/10/2014 10:46:46, 191.161.21.31, admin,
http://localhost:80/projects/83907/artifacts, POST, 401.1722, 500,
Processing of the HTTP request resulted in an exception.,
System.Web.Http.HttpResponseException
```

> **Important:** Depending on whether the log entry category is applicable to the action that occurred, the log entry field either contains data or is blank.

Here's an explanation of the data contained in each log entry, outlined in the order the columns appear:

Log Entry Data	Description	Example

TypeOfEntry	Indicates the type of action that occurred.
	
	Possible actions include:
	
	  ■ Error
	  ■ Audit
		`Error`

Date	Indicates the date that the action was logged.	`05/23/2013 14:40:45`
SourceIp	Indicates the IP address of the user performing the request.	`191.161.21.31`
Username	Indicates the user name of the user that performed the request.	`jsmith`
UriRequest	Indicates the URI that was used in the request.	`http://localhost:80/projects/83907/artifacts`
HttpMethod	Identifies the HTTP method that was performed.	`POST`
ElapsedTime	Indicates the amount of time (in milliseconds) that elapsed between the request and the response.	`401.1722`

Log Entry Data	Description	Example
Status Code	Indicates the status code that appeared in response to the request.	`500`
Message	If a message appeared, this data indicates the response message that appeared.	`Processing of the HTTP request resulted in an exception.`
Exception	If an exception occurred, this data indicates the type of exception that occurred.	`System.Web.Http.HttpResponseException`

Downloading the Blueprint log zip file

The Blueprint log zip file provides information about system usage and can be helpful for troubleshooting. The Blueprint log zip file contains four different log files (the server log, audit log, API log and the client log). For more information about the log zip file's contents, see About Blueprint logging.

To download the Blueprint log zip file:

> **Note**: If you are using Internet Explorer 8, you must enable the *automatic prompting for file downloads* security setting before you can download the file from Blueprint. To enable this setting, click **Tools** > **Internet Options** > **Security** > **Custom level...** > **Downloads** and then enable the **Automatic prompting for file downloads** option.

1. Open the *Instance Administration Console*.
2. Click **Instance Settings**.
3. Click **Logging**.
4. Click the **Download Log** button.

After you click the **Download Log** button, your browser asks you if you want to save or open the file. The `.zip` file that you download contains various `.log` files. You can open the `.log` files using Microsoft Excel, or any text editor such as Notepad.

> **Tip**: If you experience problems opening the file in Microsoft Excel, try renaming the file so it has a `.csv` file extension.

Managing active directory settings

Active directory integration allows you to leverage your active directory infrastructure to authenticate your users on behalf of Blueprint. Blueprint provides the following options for connecting to one or more active directory servers:

- If your organization consists of a single active directory server, you can use the default active directory integration as long as your Blueprint Server User (that was specified during installation) is a member of the active directory.

- If your organization consists of a single active directory server, but your Blueprint Server User (that was specified during installation) is *NOT* a member of the active directory, you can configure custom active directory integration and add a single active directory server.
- If your organization consists of multiple active directory servers, you can configure custom active directory integration and add multiple active directory servers.

You can also disable active directory integration if your organization *only* wants to create and manage users directly in Blueprint.

Configuring default active directory integration

If you use the default active directory integration, Blueprint automatically uses the active directory server that the Bluerpint Server User is a member of.

If your Blueprint Server User is not part of an active directory, or if you wish to specify multiple active directory servers, you can configure custom active directory integration.

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**.

If you need to remove an active directory server at any time, you can click the active directory server on the leftmost side of the screen and then click the **Remove** button.

Configuring custom active directory integration

Custom active directory integration allows you to specify one or more active directory servers. By specifying multiple active directory servers, you can add users to Blueprint from multiple forests.

For example, you can specify the same bind user for multiple active directory servers, but define a different LDAP URL for each server so it points to different domains.

> **Note**: When multiple active directory servers are defined, a **Connection** option appears on the Add From Windows dialog when you are adding a Windows user to Blueprint. The **Connection** option allows you to choose the active directory server that contains the user(s) that you want to add to Blueprint.

Configuration Requirements

You acquire the following information from your active directory administrator before you can configure custom active directory integration:

- BIND user SamAccountName (not the common name, as per RC2010)
- BIND user password
- AD server name (the actual server name) + fully qualified domain name
- Domain component names (that is, the full DNS name of the domain is, for example, "dc=MyDomain,dc=com")

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**.

If you need to add an active directory server at any time, you can click the active directory server on the leftmost side of the screen and then click the **Remove** button.

Disabling active directory integration

> **Warning**: If you disable active directory integration, you can no longer add Windows users to Blueprint. You are limited to adding Database users to Blueprint.

To disable active directory integration:

1. Open the *Instance Administration Console*.
2. Click **Active Directory Settings**.
3. Clear the **Enable Active Directory Integration** option.
4. Click **Save**.

Federated Authentication

Blueprint's federated authentication provides on-premise and cloud customers with the ability to leverage their existing identity provider to authenticate users in Blueprint. In other words, after a user has authenticated with your identity provider, Blueprint does not require a username and password to access the system.

What is federated authentication and SAML?

Federated authentication is the practice of allowing an external system to provide authentication services for another application. This goes beyond acting as a repository for credentials, but actually acting as the system which validates authentication attempts. One example of a federated authentication technology includes SAML.

SAML (Security Assertion Markup Language) is a technology used to implement federated authentication and single sign on (SSO). SAML provides a secure, XML-based solution for exchanging user security information between an identity provider (your company) and a service provider (Blueprint).

How it works

With federated authentication, no direct connection is required between Blueprint and the identity provider:
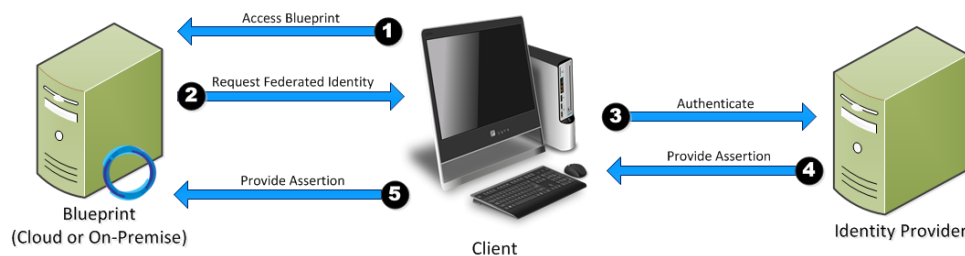
When the client accesses the service provider (that is, Blueprint), Blueprint requests that the client identifies itself through SAML. The user authenticates with the identity provider, which in turn returns an assertion (that is, a token). This token is then sent to Blueprint as proof of successful authentication and identity.

System requirements

This section outlines technology requirements and variables that are needed in order to configure federated authentication.

Federated authentication technology requirements

Blueprint supports the following federated authentication technologies:

- SAML 2.0 Token and Protocol
- Service Provider Initiated Login (Required)
- Identity Provider Initiated Login (Optional)
- SHA1 and SHA256 Signature Digests

Required variables

Identity provider requirements

- 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 the 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).

Federated authentication settings requirements

After configuring your identity provider to work with Blueprint, you must enable federated authentication in Blueprint.

You must provide information for the following fields:

- **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

Example setup

Jamal, an IT administrator, is setting up federated authentication for a company (called BP Airlines) using the Blueprint cloud instance.

First, Jamal makes sure the identity provider is configured properly, like so:

Next, Jamal configures the Federated Authentication Settings in Blueprint (*Instance Administration Console*, *Configure Settings* section). He selects **Enable Federated Authentication** and uploads a new certificate. Jamal also specifies values for the **Login URL** (defines the identity

provider service URL) and the **Logout URL** (the URL users navigate to after clicking the **Logout** button):

After the setup is complete, Jamal's chosen implementation allows cloud customers to forgo the default log-on process with the click of a link.

User flows

Service provider initiated login

1. User navigates to the Blueprint login screen.
2. User clicks the Go button.
3. User logs in with corporate identity (if not already authenticated)

The user is authenticated and can begin using Blueprint.

Identity provider initiated login

Identity provider initiated login is very flexible and may vary drastically depending on your chosen implementation. For demonstration purposes, here is a common implementation of identity provider initiated login:

1. User navigates to a company Intranet webpage.
2. User clicks a Blueprint link.
3. Blueprint is loaded and authenticated automatically.

The user is authenticated and can begin using Blueprint.

Expired session

Expired sessions can happen for both service provider initiated login and identity provider initiated login.

An expired session can happen for a variety of reasons:

- session timeout: the session has timed out due to inactivity
- session override: the user has overridden the session by logging in at a different location

Here is the typical user flow when a user encounters an expired session:

1. User is presented with a dialog explaining the session has expired
2. User clicks OK.
3. User is re-authenticated automatically, assuming the user is still authenticated with the identity provider. If the user is not still authenticated with the identity provider, the user is prompted to re-authenticate with the identity provider.

The user is re-authenticated and can continue using Blueprint.

Configuring your identity provider for Blueprint federated authentication

Before you can use Blueprint federated authentication, your identity provider must be configured properly so the token submitted to Blueprint includes all of the required information in the proper format.

> **Note**: The configuration terminology may vary slightly depending on the identity provider you are using. For example, some identity providers may use the term *Claims* instead of *Attributes*.

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 the 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).

Enabling Blueprint federated authentication

After configuring your identity provider to work with Blueprint, you must enable federated authentication in Blueprint.

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**.

About fallback from federated authentication

Fallback from federated authentication allows users to login to Blueprint using a username and password in addition to federated authentication. Blueprint supports both database and Windows (that is, LDAP) authentication when authenticating a user in fallback mode.

> **Tip**: We recommend that at least one instance administrator has this option enabled. If your federated authentication fails for any reason (example: expired certificate), this user will be able to login to Blueprint with a username and password to fix the issue.

Any number of users can be configured for fallback. When federated authentication is enabled, all users (by default) are enabled for fallback authentication. However, the user cannot login to Blueprint using the fallback method unless a password is configured for the user. When fallback is enabled, a password must be explicitly set for the user.

How do I enable 'fallback from federated authentication'?

This option can be enabled and disabled on the *Users* tab in the Instance Administration Console. The setting is called **Allow fallback from federated authentication**. This setting must be configured for each user.

The fallback authentication only appears as a option for users when federated authentication is enabled.

Managing instance-level office document templates

Office document templates can be added at the instance level. Once added, users can generate office documents using the data stored in Blueprint artifacts.

Here are the typical sequence of events:

1. Instance administrator authors a new template
2. Instance administrator adds the template to Blueprint
3. Users generate documents using the templates

Adding an office document template to an instance

After you add an office document template, users can generate documents.

To add a new document template:

1. Open the *Office Document Templates* tab.
   1. Open the *Instance Administration Console*.
   2. Click **Office Document Templates** (*Office Integration* group).
2. Click the **New** button on the ribbon (*Actions* group).

3. Configure the settings for the new template:

- **Include Files**: If selected, the generated document is packaged in a `.zip` file with the documents and attachments.
- **Rich Text Formatting**: If selected, any rich text formatting in Blueprint is preserved in the generated document.You can also select the percentage you want to scale font size for rich text fields in the generated document.
- **Include Open Discussions**: If selected, open discussions are included in the generated document.
- **Include Closed Discussions**: If selected, closed discussions are included in the generated document.
- **Attach source data to report output**: If selected, the generated document is packaged in a `.zip` file with the project XML data that was used to generate the document.

  The project XML data can be useful if:

  - you want to author a template using data and image references from your project.
  - you are debugging a problem with the template you are using
  - you want to provide Blueprint support with the project XML data that generated the template

4. Upload the office document template that you created.
   1. Click the **Upload** link.
   2. Locate the file that you want to upload.
   3. Click **Open**.
5. Click **Save**.

Users can now select this template when they generate office documents.

About job services

A job service is an instance-level component that provides Blueprint users with the ability to execute specific jobs, such as document generation. Blueprint allows Instance Administrators to view the statuses of all job services in the instance. In the event that there is a problem processing jobs, the Instance Administrator can check the status of the job service(s) and troubleshoot accordingly.

The *Job Services* tab provides the following information about each job service:

- **Service Name**

  Although it is not recommended, the job service name can be modified in a configuration file.

- **Supported Jobs**

  A job service can be configured to only allow execution of specific types of jobs. For example, an administrator could configure the job service to only support document generation.

  > **Note:** When a job service is configured to only allow certain jobs, any jobs that are not supported by the configuration still appear in the queue after being started.

- **Last Active Time**

  The last date and time that a service was active. If the service is not responding, this information may be helpful for troubleshooting purposes.

- **Status**

A job service can have any of the following statuses:

- Stopped
- Idle
- Active
- Not Responding

- **Current Job ID**

The job ID appears if the job is active.

To check the status of a job service:

1. Open the *Instance Administration Console*.
2. Click the **Job Services** link.

The *Job Services* list appears in a new tab.

Managing e-mail settings

Overview

E-mail settings must be configured in Blueprint if you want to take advantage of Blueprint review notifications, e-mail-integrated discussions or the ability to mention someone in a comment. For more information on review notification settings and e-mail integration discussion settings, see About review notification settings and About e-mail integration discussion settings.

About review notification settings

Blueprint review notifications provide your users with information and reminders at key moments. Blueprint offers the following review e-mail notifications:

- Review Start: A review notification is sent to all review participants when a review is started.
- Review Close: A review notification is sent to all review participants when a review is closed.
- Review Participant Removal: A review notification is sent to the user when the user is removed from a review.

> **Note**: E-mail settings are dependent on your company's e-mail server configuration. Contact your IT department to obtain the proper settings.

Enabling and configuring review notifications

To take advantage of review notifications, you must:

- enable review notifications and configure e-mail settings
- ensure that each user has an associated e-mail address

Important: Notifications are not sent to users who do not have an associated e-mail address.

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.

About e-mail integration discussion settings

Within comments you can mention other users as well as individuals that do not have Blueprint licenses. Whenever a reference to a user or an e-mail address is made within a comment, a notification message is sent to the user or e-mail account. Mentioning stakeholders in discussions can allow them to contribute to the discussions via e-mail.

Tip: E-mails can also be sent manually if a user wants to share an artifact using e-mail.

Note: E-mail settings are dependent on your company's e-mail server configuration. Contact your IT department to obtain the proper settings.

Enabling and configuring e-mail integrated discussion settings

To take advantage of e-mail-integrated discussions, you must:

- enable e-mail integrated discussions and configure the settings
- ensure the option to have discussions via e-mail is enabled within Project Settings (Project Administration Console)

> **Important:** In enabling e-mailed integrated discussions, you are consequently permitting information within Blueprint to be made external to Blueprint via the medium of e-mail. Anyone with an e-mail address that is mentioned in a comment receives an e-mail message containing a snapshot of the associated Blueprint artifact. We recommend that administrators address and/or discuss any security concerns with the appropriate parties before enabling this feature.

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* (Project Administration).

Modifying the default print template at the instance level

Blueprint provides instance administrators with a default Word template for the purpose of exporting and printing artifacts.

The template is written and designed using the document template authoring add-in.
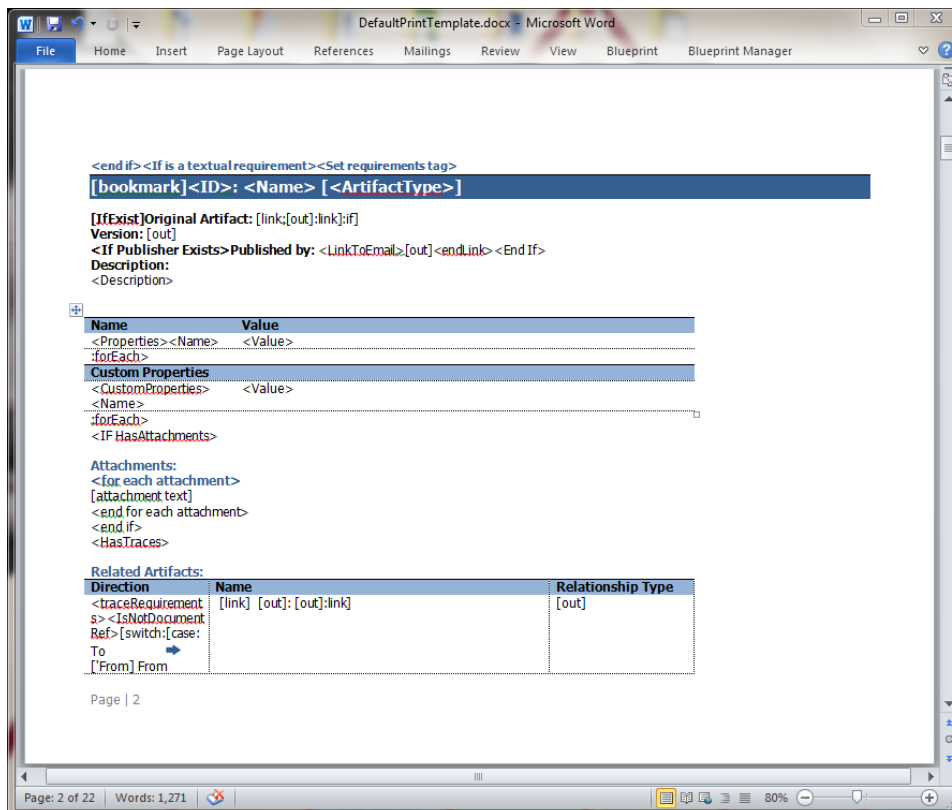
We do not recommend making changes to the template directly unless you have document template authoring experience. For more information about document template authoring, see the Document Template Authoring Help.

> **Note:** The default print template can be changed at two different levels of administration privileges. Instance administrators with the applicable privileges can modify the default print template within the *Instance Administration Console*, setting a new default template for all projects. Both instance administrators and project administrators with the applicable privileges can change the default print template within the *Project Administration Console*, which sets a new print template for that individual project only.

Changes that are made to the template at the project level override the instance level template within the specific project only.

To modify the default instance print template:

1. Open the *Instance Administration Console*.

2. Click the *Instance Print Template* link.

3. Create a new template or modify an existing document template.

    If you want to modify the existing template, click the **Download** link.

    > **Note**: If you are using Internet Explorer 8, you must enable the *automatic prompting for file downloads* security setting before you can download the file from Blueprint. To enable this setting, click **Tools** > **Internet Options** > **Security** > **Custom level…** > **Downloads** and then enable the **Automatic prompting for file downloads** option.

    > **Note:** When you replace the default print template, you are setting a new default template for all individual projects.

4. Click the **Replace** link.

    The *save* dialog box appears.

5. Select your new print template and then click **Open**.
6. Click **Save**.

Your new instance print template is saved. Whenever you click the **Print to PDF** button or the **Print to Word** button on the ribbon (*Home* tab), your print template is used to export an artifact to a file for printing purposes.

To restore the system default document template, click **Restore** and then click **Save** within the *Instance Print Template* screen.

About Accelerators

> **Note:** Only instance administrators can install Packs.

Accelerators can be used to automatically configure Blueprint to suit particular standards, processes or software development approaches.

An Accelerator is a zip file that contains one or more components used to pre-define content, such as standard artifact types, standard artifact properties, custom artifact types, custom artifact properties, type-property mappings, project folder structure, pre-defined artifacts and other data.

The components containing the pre-configured content may be any of the following:

- Blueprint Pack (`.bpk` file)

  Contains pre-configured data, such as artifact types, properties or glossary terms, to help users accelerate the development of requirements. For example, the agile pack contains *epic*, *user story*, *theme* and *feature* artifact types for users that are developing requirements using an agile approach.

  > **Note:** The Agile Pack is automatically installed by default on Blueprint versions 6.1 and later.

- Blueprint project

  A project containing pre-configured content such as project folder structure, custom artifact types, custom artifact properties and/or artifact content.

- A Microsoft Excel workbook (`.xlsx` file) containing requirements artifacts

To install a Pack:

1. Open the *Instance Administration Console*.
2. Click the **Install Pack** link.

   The *Install Pack* page appears in a new tab.
3. Click the ⋯ button.

   The *Open* dialog appears.
4. Select the Pack you want to install and then click **Open**.
5. Click **Install**.

   > **Note:** Pack installation may take several minutes.

The Pack has been successfully installed.