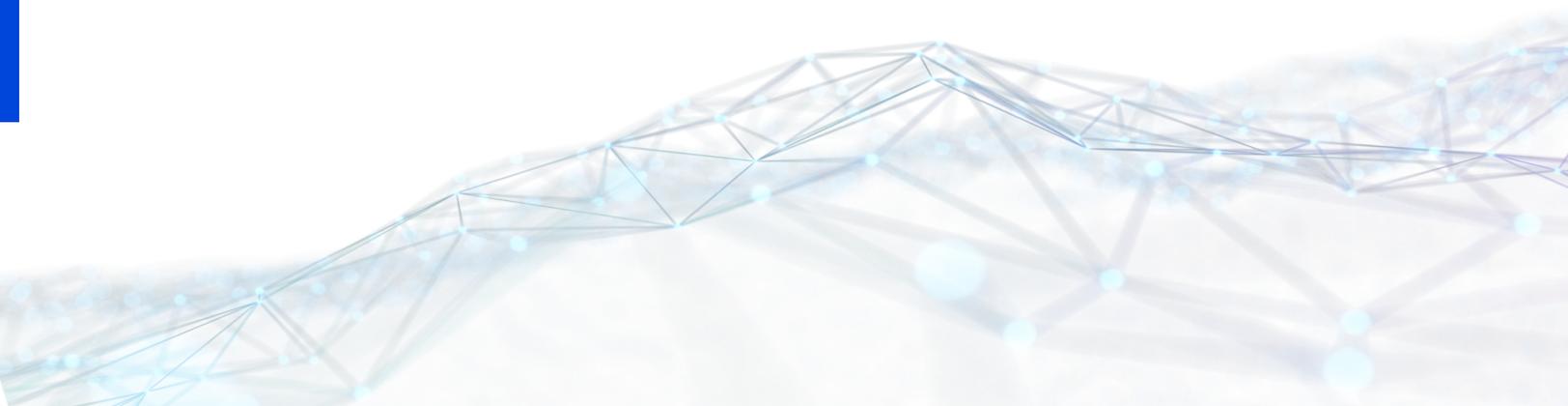




TIBCO® BPM Enterprise

Administration

Version 5.6.0 | November 2024



Contents

Contents	2
Introduction to Administration	5
Container Orchestration System Administration - Kubernetes	7
readinessProbe	7
livenessProbe	8
Injected Configuration	8
TIBCO BPM Enterprise Administrator	10
Accessing Administrator	10
Configuring the Admin User	11
Manage Client Applications	11
Customizing your Application	13
Deployment Manager	14
Viewing Deployed Applications	14
Deploying Applications	15
Undeploying Applications	16
Shared Resources	17
HTTP Client Shared Resources	18
KeyStore Provider Shared Resources	24
SSL Client Provider Shared Resources	26
SMTP Connection Shared Resources	30
SAML Authentication Shared Resources	32
OpenID Authentication Shared Resources	39
JDBC Connection Shared Resources	48
Organization Browser	55
Accessing the Organization Browser	55

Browsing the Organization Model	56
LDAP Containers	65
Resources	98
Privileges	110
Capabilities	112
Resource Attributes	115
Push Destinations	117
Dynamic Organization Model Extension Points	122
Candidate Queries	131
Process Manager	147
Process Manager Columns	147
Customizing Process Manager Columns	148
Process Instance State	148
Searching for Process Instances	149
Filtering Process Templates	150
Starting a Process Instance	151
Suspending a Process Instance	152
Canceling a Process Instance	153
Fixing a Process Instance	154
Process Instance Migration Overview	155
Purging Process Instances	156
Contribute Your App	157
Configuration Management	158
Authentication Configuration	159
Case Data Management Configuration	160
Directory Engine Configuration	161
Event Collector	163
Event Publication Configuration	167
Logging	169
Pageflow Engine Configuration	170
Process Engine Configuration	170
Work Presentation Configuration	173

Authentication	174
SAML Web Profile Authentication	175
OpenID Connect Authentication	176
System Actions	179
Scope of System Actions	180
Example of using System Actions to Control Users' Access to System Functions ..	182
System Actions and Organization Model Versions	187
List of Messages	190
Auditable Messages	190
Message Categories and Attribute Contents	190
Container Management	192
Database Support	194
Appendix: Utility Commands	195
TIBCO Documentation and Support Services	199
Legal and Third-Party Notices	201

Introduction to Administration

There are several methods of administering TIBCO® BPM Enterprise.

- From the container orchestration system - The container orchestration system provides probes for determining a container's readiness and health, as well as tools to administer the scaling of pods based on metrics from TIBCO BPM Enterprise. These are administered from the container orchestration system.

In this release of TIBCO BPM Enterprise, the only supported container orchestration system is Kubernetes.

For more information, see [Container Orchestration System Administration - Kubernetes](#).

- From the TIBCO BPM Enterprise Administrator - TIBCO BPM Enterprise provides an Administration feature that can be used to manage deployed applications, configure shared resources, manage organizations and resources, and so on.

For more information, see [TIBCO BPM Enterprise Administrator](#).

- Using the TIBCO BPM Enterprise REST API - There are two administration-related services that are publicly available to administer a TIBCO BPM Enterprise system:
 - Administration Service - This service is used to create, read, update, and delete shared resources, as well as configure Single Sign-On (SSO) authentication used by the TIBCO BPM Enterprise containers.
 - Property Management Service - This service is used to create, read, update, and delete properties used by TIBCO BPM Enterprise components.

Information about these services can be found in the TIBCO BPM Enterprise API Explorer.

1. Access the API Explorer using the following URL (for information about the privileges needed to access API Explorer, see "Access to Administration" below):

```
protocol://host:port/apps/login
```

where:

- *protocol* is the communications protocol being used, either http or

https. This was specified at installation.

- *host* is the DNS name or IP address of the server hosting the TIBCO BPM Enterprise runtime.
 - *port* is the port being used. The default port is 80.
2. Log in with a valid TIBCO BPM Enterprise username and password.
 3. Click .
 4. Click **API Explorer**.

Access to Administration

You need Administrator-level credentials to perform administration tasks in TIBCO BPM Enterprise.

- TIBCO BPM Enterprise Administrator - To access the Administrator in TIBCO BPM Enterprise, you must either be in the System Administrator group in Organization Model Version 0 (see [Organization Model Version 0](#)), or have the system action, System Administration.
- Application Development - To access Application Development in TIBCO BPM Enterprise, you must have the Application Development system action assigned to you.
- REST API - To access the TIBCO BPM Enterprise Administration Service and Property Management Service, you must be assigned the Application Development system action.

For more information, see the "System Actions Reference" topic in the *TIBCO Business Studio - BPM Edition Application Designer's Guide*.

Container Orchestration System Administration - Kubernetes

The types of TIBCO BPM Enterprise-related administration that can be performed from Kubernetes are described in the following subsections. These configurations are specified in the Kubernetes deployment file.

readinessProbe

TIBCO BPM Enterprise uses the Kubernetes readinessProbe to determine a container's readiness to accept work.

Kubernetes can perform an HTTPGetAction on the following REST endpoint to probe for readiness:

```
path: /bpm/adapter/v1/readiness
port: <port_used_for_BPM_services>
```

TIBCO BPM Enterprise returns one of the following responses:

- HTTP 200 - The TIBCO BPM Enterprise container is ready to accept requests.
- HTTP 503 - The TIBCO BPM Enterprise container is not ready to accept requests.

The readiness REST endpoint has no required authentication.

The following parameters can be defined with the readinessProbe:

- `initialDelaySeconds` - Number of seconds after the container has started before readiness checks are sent.
- `periodSeconds` - How often, in seconds, to perform the probe.
- `timeoutSeconds` - Number of seconds after which the probe times out if no response is received.
- `successThreshold` - Minimum consecutive successes for the probe to be considered successful after having failed.

- `failureThreshold` - If a pod starts and the probe initially fails (HTTP 503 response), Kubernetes will re-try the probe this number of times before giving up.

livenessProbe

TIBCO BPM Enterprise uses the Kubernetes `livenessProbe` to ensure that the container is healthy and responsive.

Kubernetes can perform an `HTTPGet` on the following REST endpoint to probe for liveness:

```
path: /bpm/adapter/v1/Liveness
port: <port_used_for_BPM_services>
```

TIBCO BPM Enterprise returns one of the following responses:

- HTTP 200 - The TIBCO BPM Enterprise container is healthy and responsive.
- HTTP 503 - The TIBCO BPM Enterprise container is not healthy. This same response is assumed if the container fails to respond.

The liveness REST endpoint has no required authentication.

The following parameters can be defined with the `livenessProbe`:

- `initialDelaySeconds` - Number of seconds after the container has started before liveness checks are sent.
- `periodSeconds` - How often, in seconds, to perform the probe.
- `timeoutSeconds` - Number of seconds after which the probe times out if no response is received.
- `successThreshold` - Minimum consecutive successes for the probe to be considered successful after having failed.
- `failureThreshold` - If a pod starts and the probe initially fails (HTTP 503 response), Kubernetes will re-try the probe this number of times before giving up.

Injected Configuration

The configuration that must be present for the TIBCO BPM Enterprise container to be started is injected into the pod using environment variables.

The following are the environment variables that are injected into the pod before container startup:

SSL Configurations

The following SSL Configuration environment variables are provided in the deployment samples that are included with the TIBCO BPM Enterprise installer.

Parameter	Description
JDBC_SSL_CONFIG	<p>This is the SSL configuration for the JDBC connection. It is essentially a direct passthrough to the JDBC driver. A basic PostgreSQL example is:</p> <pre>ssl=true;sslmode=verify-full;sslrootcert=<full_path_to_ssl_root_certificate></pre> <p>Since this is a passthrough to the driver, you can look at the PostgreSQL JDBC driver specification for more details.</p> <p>Note that it's important that the <full_path_to_ssl_root_certificate> is the full path as seen from inside the container. Essentially, it involves getting the SSL certificate mounted into the container on a specific path, which is then used as <full_path_to_ssl_root_certificate>.</p>
LDAP_<GROUP_NAME>_SSLCERT	<p>This is for LDAP SSL, where <i>GROUP_NAME</i> is the same as the other LDAP environment variable (see above). For example, if you have LDAP_SYSTEM_ALIAS defined for an LDAP Directory Connection, this would be LDAP_SYSTEM_SSLCERT. Its value is simply the full path to the certificate required to connect to the LDAP server, much like it is for JDBC. The difference here is that it is the only component of the value, so an example value is simply:</p> <pre><full_path_to_ldap_ssl_certificate></pre> <p>You would make this certificate available to the container the same way as you would for JDBC.</p>

TIBCO BPM Enterprise Administrator

TIBCO BPM Enterprise provides an Administrator that is used to configure various elements of a TIBCO BPM Enterprise system.

The following subsections provide information about the administration tasks that can be performed from the Administrator.

Accessing Administrator

To access the Administrator in TIBCO BPM Enterprise, you must either be in the System Administrator group in Organization Model Version 0 (see [Organization Model Version 0](#)) or have the System Administration system action.

Procedure

1. Enter the following URL in your browser:

```
protocol://host:port/apps/login
```

where:

- *protocol* is the communications protocol being used, either http or https. This was specified at installation.
 - *host* is the DNS name or IP address of the server hosting the TIBCO BPM Enterprise runtime.
 - *port* is the port being used. The default value is 80.
2. Log in with a valid TIBCO BPM Enterprise username and password.
 3. Click .
 4. Click **Administrator**.

Configuring the Admin User

TIBCO BPM Enterprise is delivered with one user already defined. This internal user is the only one authorized to log in until another user is configured.

The default name of this internal user is **tibco-admin**, and the LDAP Distinguished Name (DN) for this user is by default set to **uid\=admin,ou\=system**.

You can change the admin login name and the LDAP entry that the admin user references.

Procedure

1. Make sure that an LDAP directory on the LDAP server contains the user that you want to use as the **tibco-admin** user.
2. Make sure that the user you want to use as the **tibco-admin** user has at least read access to each LDAP directory that you intend to use.
3. Use the provided "utility" Docker image to configure the **tibco-admin** LDAP user.
4. Usage:

```
docker run -it --rm tibco/bpm/utility:5.6.0 utility -setupAdminUser
```

Manage Client Applications

The **App Development** page is used to browse, customize, save, preview, and publish applications. Applications are hosted on the platform, so when users publish them, they are available to other users.

i Note: A number of third-party common libraries are available for use under `app_cdn`.

You can do the following:

- Upload an application. In the **New Upload** tab, you can upload the ZIP file of your application. The uploaded application is shown in the **All Uploads** tab.
- Edit application files (**Browse**): Access the application files in a file editor and make

the changes. You can then save, test, and publish the application to view the changes. Every time you edit an application and save it a new version of the application is saved. You can view the latest version number of the application in the **Current Version** column. When you are publishing an application, you can use these versions. Multiple users can access applications, but cannot merge changes concurrently.

- Clone an application (**Clone**). After cloning an application, update the `app.desc.json` file.
- Delete an application (**Delete**).
- Download an application (**Download**). You can work offline and reupload them.
- View application details like owner, created time, versions (**Details**). Here, you can select a previous version of the application to view it.
- Publish an application (**Publish**).
- Launch an application (**Launch**): You can launch either the latest application or a previously published version of the application.
- The following properties can be passed to an application from the `appName.app.desc.json` file:



Caution: For the properties marked as mandatory, you can modify their value, but do not unset the property, otherwise the application might not function properly.

- `appImage` (mandatory for applications targeted for the mobile devices. Not applicable to the desktop applications.) - The path to the image depicting the application.
- `category` (optional) - A category used to group your applications.
- `configPage` (mandatory) - The path to the configuration file for the application.
- `defaultLocale` (optional) - The locale in which the application is displayed.
- `description` (mandatory property for applications targeted for mobile devices.) Short description for the application.
- `errorPage` (optional) - The path to the error page for the application.
- `indexPath` (mandatory) - The path to `index.html` file for the application or the

landing page for the application.

- `library` (optional: default=false) - Boolean flag that describes if the application is a library that can be used by other applications.
- `loginPage` (mandatory) - Application is directed to this page when the user logs in.

i Note: The out-of-the-box example application in Application Development uses the login component provided by the framework: `/apps/logincomponent/login.html`. Therefore, you have the option of using that same login component, or redirect to a custom login page using this property.

- `mobile` (optional: default=false) - Boolean flag that describes if the application is for mobile devices.
- `support` (optional) - Languages supported by mobile applications.

⚠ Caution: Removing any of the mandatory properties might prevent the application from operating correctly.

Customizing your Application

You can customize an application by changing its logo, title, fonts, and font and header color.

Procedure

1. Enter the following URL in your browser:

```
protocol://host:port/apps/login
```

where:

- *protocol* is the communications protocol being used, either `http` or `https`. This was specified at installation.
- *host* is the DNS name or IP address of the server hosting the TIBCO BPM

Enterprise runtime.

- *port* is the port being used. The default value is 80.
2. Log in with a valid TIBCO BPM Enterprise username and password.
 3. From the toolbar, select the paint easel icon .

 **Note:** The paint easel icon is available only if you have the System Administration system action.

4. Customize the application logo, title, font, and font and header color, as desired.
You can **Reset Theme** if you want to return to the original theme.
5. Click **Save**.

Deployment Manager

You can use Deployment Manager to deploy, upgrade, and undeploy applications that have been designed in TIBCO Business Studio - BPM Edition to a TIBCO BPM Enterprise server.

An application is deployed to TIBCO BPM Enterprise by deploying TIBCO Business Studio - BPM Edition projects to a TIBCO BPM Enterprise server.

For more information about deployment, see "Application Deployment" in *TIBCO Business Studio - BPM Edition Application Designer's Guide*.

Viewing Deployed Applications

From Deployment Manager, you can see a list of your currently deployed applications. Select **All Deployments** to view information about your applications, like their status, version, and deployment time.

An application can have one of the following states:

State	Description
Deployed	The application is deployed and you can now start an instance of a process

State	Description
	template.
Undeploying	The application is currently undeploying. An application may fail to undeploy because there are outstanding process instances or work items. Once an application has undeployed, it is no longer displayed in All Deployments .

Deploying Applications

Once your TIBCO BPM Enterprise project has been generated as a deployment artifact, you can deploy it using Deployment Manager.

Before you begin

The TIBCO BPM Enterprise project has been generated as a deployment artifact.

i Note: The projects need to be deployed in order of dependency. For example, you must deploy a business data project before the process projects that reference it. Similarly, you must deploy projects with sub-processes before projects with the main processes that invoke these. If you don't, then the deployment fails and you need to deploy the referenced projects in the correct order.

Procedure

1. Select **New Deployment**.
2. Either drag the RASC file to **New Deployment** or click **Select RASC file** to browse to its location.

A TIBCO BPM Enterprise project is deployed, if it is:

- A new application
- A later version of an existing application.

3. Select **Deploy**.

After the project has deployed, its status is shown as **Deployed** in **All Deployments**.

The Type column also shows you the type of project deployed - Process, Organization, Data, and so on.

Undeploying Applications

You can undeploy a deployed application.

Click  for the application you want to undeploy and select **Undeploy**. Once the application has undeployed, it is no longer displayed in **All Deployments**.

An application may fail to undeploy because there are outstanding process instances or work items. If after clicking **Undeploy**, the application goes into an Undeploying state and does not change, follow the steps given below:

1. Click  for the application, which is now in Undeploying state, and click **Details**.
2. A small pane appears to the right and shows the list of active instances that are preventing the undeployment of the application.
3. Click Process Instances. The name of the process instance is displayed in the pane with  on the right.
4. Click  for process instances.
5. Click **Purge** in the dropdown.
6. Click **Yes, purge** in the **Purge Instances confirmation** dialog.
7. Click  for the application, which is now in *Undeploying* state, and select **Undeploy**.
8. Click **Yes, undeploy** in the **Warning** dialog box.

 **Note:** A data application remains in the *undeploying* state if there are any active cases associated with it. These active cases can be viewed and purged from application details.

Once the application has undeployed, it is no longer displayed in **All Deployments**.

Alternatively, you can cancel, or complete, any outstanding process instances or work items for that application. In this situation, the **Deployment Manager** displays the number of active instances, as well as the application type. Within a few minutes of the cancel or

complete operation of all the active instances, the **Deployment Manager** will complete the undeployment.

i Note: Projects need to be undeployed in order of dependency. For example, you must undeploy a process project before the business or case data project that references it.

Shared Resources

Shared resources contain connection details for physical resources. Shared resources provide a way to use an identifier to reference the configuration of a resource, allowing multiple applications to use an instance of the shared resource without having to configure the resource multiple times.

The following shared resources can be configured in TIBCO BPM Enterprise:

- [HTTP Client Shared Resources](#) - These specify connection details for REST service activities in BPM applications.
- [SSL Client Provider Shared Resources](#) - These maintain credentials needed by SSL clients.
- [KeyStore Provider Shared Resources](#) - These specify details about a KeyStore, which can be used to establish a secure connection via SSL.
- [SMTP Connection Shared Resources](#) - These are used by component implementations to send and receive messages to and from SMTP mail servers.
- [SAML Authentication Shared Resources](#) - These are used for SAML Web Profile authentication, which allows users of your application to log in using a username and password issued by an Identity Provider (IdP) that supports SAML Web Profile.
- [OpenID Authentication Shared Resources](#) - These are used for OpenID Connect authentication for both inbound and outbound calls.
- [JDBC Connection Shared Resources](#) - These are used by database activities in TIBCO BPM Enterprise applications to access databases.

HTTP Client Shared Resources

HTTP Client shared resources specify connection details for REST service activities in BPM applications.

BPM applications are bound to an HTTP Client shared resource when the application is deployed.

HTTP Client shared resources can be created, edited, or deleted using the TIBCO BPM Enterprise Administrator.

Creating or Editing an HTTP Client Shared Resource

HTTP Client shared resources can be created or edited using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **Http Clients**.
3. Click  to add a new shared resource, or select an existing shared resource to edit it.
4. Configure the shared resource using the following descriptions.

Definition

Property	Description
Name	(Required) The unique name of the HTTP Client. The name value is case insensitive.
Description	A description for the HTTP Client.
Machine name	(Required) The name of the host or the third-party server that is being accessed.
Port	(Required) The port number on which to invoke outgoing HTTP requests.

Property	Description
Enable SSL	Select to enable SSL. When selected, an SSL Client Provider name must be specified in the Ssl client provider name field. See SSL Client Provider Shared Resources . Default: <i>false</i>
SSL client provider name	(Required if Enable SSL is selected.) The name of an SSL Client Provider.

Timeout and socket

Property	Description
Socket timeout	The maximum amount of time (in milliseconds) the socket waits for data, or period of inactivity between consecutive data packets, before timing out. This should be changed when connecting to very slow external services. A timeout value of zero is interpreted as an infinite timeout. Default: 0
Connection timeout	The number of milliseconds to wait for a connection to be established before timing out. This should be changed when connecting to very slow external services. A timeout value of zero is interpreted as an infinite timeout. Default: 0
Stale check validation	Specifies how a stale connection check is applied. When set to a positive value, a stale connection check is made if the inactivity time exceeds the time set (in milliseconds), otherwise, no stale connection check is made. Disabling the stale connection check can result in slight performance improvement at the risk of getting an I/O error when running a request over a connection that has been closed on the server side. Default: -1

Property	Description
Time to live	<p>The maximum time, in milliseconds, that a connection remains available for use. When set to a positive value, a connection that is older than the time set will no longer be available for use and be replaced with a new connection, otherwise, the connection age is not checked.</p> <p>Default: -1</p>
Buffer size	<p>Socket buffer size, in bytes.</p> <p>This is a suggestion to the kernel from the application about the size of the buffers to use for the data transferred over the socket.</p> <p>Default: -1 (allow the runtime to determine the buffer size)</p>
Local socket address	<p>A local host address to be used for creating the socket.</p>
Reuse address	<p>When a TCP connection is closed, the connection might remain in a timeout state for some time after the connection is closed (typically known as the TIME_WAIT state or 2MSL wait state).</p> <p>For applications using a well-known socket address or port, it might not be possible to bind a socket to the required SocketAddress if there is a connection in the timeout state involving the socket address or port.</p> <p>Setting this option to true allows the socket to bind to an address that is still in a TIME_WAIT state.</p> <p>Default: false</p>
Suppress TCP delay	<p>Specifies whether the Nagle algorithm is used.</p> <p>The Nagle algorithm tries to conserve bandwidth by minimizing the number of segments that are sent. When applications want to decrease network latency and increase performance, they can disable Nagle's algorithm using this property.</p>

Property	Description
	Data is sent earlier at the cost of an increase in bandwidth consumption and the number of packets. Default: true

Redirections

Property	Description
Accept redirect to different host	This controls if an HTTP redirection can redirect to a different host. Also see 'Accept redirect to different port ' and 'Accept redirect to HTTP '. Each of these properties can be independently set to allow whatever combination is required.
Accept redirect to different port	This controls if an HTTP redirection can redirect to a different port.
Accept redirect to HTTP	This controls if an HTTP redirection can redirect from HTTPS to HTTP. Note that the reverse is not possible. If the HttpClient was set up for HTTP, then it will not contain a SslClientProvider that would allow redirection from HTTP to HTTPS.

Proxy

Property	Description
Configure proxy	Select to configure the HTTP Proxy options.
Proxy type	(Required if Configure proxy is selected.) Type of proxy server. The available selections are: HTTP or SOCKS V4/V5
Proxy host	(Required if Configure proxy is selected.) The address of the proxy host.

Property	Description
Proxy port	(Required if Configure proxy is selected.) The port of the proxy host.
Configure proxy basic authentication	Select to configure access to the proxy server with a username and password.
Proxy username	(Required if Configure proxy basic authentication is selected.) The username used for the proxy server basic authentication.
Proxy password	The password used for the proxy server basic authentication.

Authentication

Property	Description
Configure Authentication Type	<p>Select from the following list:</p> <ul style="list-style-type: none"> • Basic authentication: To configure access to the proxy server with a username and password. • OpenID authentication: To configure access to the OpenID authentication server.

Property		Description
Basic authentication	Realm	The realm that contains the user information.
	Username	(Required if Configure basic authentication is selected.) The username used for the proxy server basic authentication.
	Password	The password used for the proxy server basic authentication.
	Preemptive authentication	If enabled, the HTTP Client uses preemptive authentication.
OpenID authentication	OpenID shared resource name	Enter the name of the newly created OpenID shared resource.

5. Click **Save**.

After saving the HTTP Client Shared Resource configuration, you can return to the **Definition** tab and click **Test connection** to test the shared resource. You must know the target endpoint to test the connection.

Deleting an HTTP Client Shared Resource

An existing HTTP Client shared resource can be deleted using the TIBCO BPM Enterprise Administrator.

 **Note:** An HTTP Client shared resource that is in use by a BPM application cannot be deleted.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **Http Clients**.
3. From the **Shared Resources** list, select the HTTP Client shared resource you want to delete.

Note: If the selected HTTP client shared resource is referenced by any applications, a list of those applications is shown on the **Definitions** tab.

4. Click the  icon, then select **Delete**.

KeyStore Provider Shared Resources

If you configure your environment for SSL, you must also set up a KeyStore. As part of the process, you configure a KeyStore provider.

SSL uses keys and certificates when it establishes a secure connection. A *KeyStore* is a database of keys and certificates. A KeyStore password is required to access or modify the KeyStore. Access to KeyStores is provided by a KeyStore Provider shared resource.

KeyStore Provider shared resources can be created, edited, or deleted using the TIBCO BPM Enterprise Administrator.

Creating or Editing a KeyStore Provider Shared Resource

KeyStore Provider shared resources can be created or edited using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **Keystore Provider**.
3. Click .
4. Configure the KeyStore Provider shared resource using the following descriptions.

Definition

Property	Description
Name	(Required) The unique name of the KeyStore Provider shared resource. The name is case insensitive.

Property	Description
Description	A description of the KeyStore Provider shared resource.
Select Keystore type	<p>(Required) The type of KeyStore. The available selections are:</p> <ul style="list-style-type: none"> • JKS - This 'Java KeyStore' is provided by Sun in the standard JDK. • JCEKS - This 'Java Cryptography Extension KeyStore' is available if you have Sun's JCE (Java Cryptography Extension) installed. This KeyStore provides much stronger protection for stored private keys by using Triple DES encryption. • PKCS12 - This 'Public-Key Cryptography Standards' KeyStore provides a format for storing server certificates and private keys in a single encrypted file. • Other - Select this if the KeyStore is something other than what is provided in the down-drop list. Specify the KeyStore type in the Enter Keystore type field.
Enter Keystore type	Specify the KeyStore type if <i>Other</i> is selected in the Select Keystore type field.

Keystore

Property	Description
Upload Keystore file	<p>Use this button to upload a keystore file to TIBCO BPM Enterprise.</p> <p>If the keystore file is successfully uploaded, "Keystore uploaded" is displayed, and the name of the uploaded file is shown.</p>
Security Provider	<p>The name of a KeyStore security provider. For example:</p> <ul style="list-style-type: none"> • Sun - Specify this if you are using the JKS KeyStore type. • SunJCE - Specify this if you are using the JCEKS KeyStore type. • SunJSSE - Specify this if you are using the PKCS12 KeyStore type.

Property	Description
	If not specified, the JVM default based on the specified KeyStore type is used.
Password	The password used to unlock the KeyStore.

5. Click **Save**.

Deleting a Keystore Provider Shared Resource

An existing KeyStore Provider shared resource can be deleted using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **Keystore Provider**.
3. From the **Shared Resources** list, select the KeyStore Provider shared resource you want to delete.
4. Click  , then select **Delete**.

SSL Client Provider Shared Resources

SSL Client Provider shared resources maintain credentials needed by SSL Clients.

An SSL Client Provider shared resource must be configured if SSL is specified for an HTTP Client shared resource. See [HTTP Client Shared Resources](#).

SSL Client Provider shared resources can be created, edited, or deleted using the TIBCO BPM Enterprise Administrator.

Creating or Editing an SSL Client Provider Shared Resource

SSL Client Provider shared resources can be created or edited using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **SSL Client Provider**.
3. Click .
4. Configure the SSL Client Provider shared resource using the following description.

Definition

Property	Description
Name	(Required) The unique name of the SSL Client Provider. The name value is case insensitive.
Description	A description for the SSL Client Provider.
Trust store provider name	(Required) The name of the KeyStore Provider to use as the Trust Store. See KeyStore Provider Shared Resources .
Security Provider	The name of an SSL Security Provider. For example, <i>SunJSSE</i> . If not specified, the JVM default is used.
Verify Remote Hostname	Selecting this option causes the name on the server's certificate to be verified against the server's hostname. If the server's hostname is different than the name on the certificate, the SSL connection will fail. The name on the certificate can be verified against another name by providing a hostname in the Expected Remote Hostname field.
Expected Remote Hostname	(Required if Verify Remote Hostname is selected) The expected hostname value to check. If a value is specified, that value is used to verify the SSL hostname, otherwise, the default SSL hostname verification is used.

Mutual authentication

Property	Description
Enable Mutual Authentication	Select this option if the SSL Client will authenticate to the server. Selecting this option causes the following three fields to become enabled.
Identity Store provider name	(Required if Enable Mutual Authentication is selected) The name of the KeyStore Provider containing the identity used for mutual authentication.
Key alias for identity	(Required if Enable Mutual Authentication is selected) The alias name for the identity used for mutual authentication.
Key Alias Password	The password for the identity used for mutual authentication.

SSL and Ciphers

Property	Description
SSL Protocol	<p>The SSL protocol used. The available selections are:</p> <ul style="list-style-type: none"> • TLS_V1 • TLS_V1.1 • TLS_V1.2 • TLS_V1.3 <p>Default: TLS_V1.2</p>
SSL Cipher Class	<p>The SSL cipher class, which specifies the number of bits in the key used to encrypt data. The greater the number of bits in the key (cipher strength), the more possible key combinations and the longer it would take to break the encryption. The available selections are:</p> <ul style="list-style-type: none"> • ALL_CIPHERS • AT_LEAST_128_BITS

Property	Description
	<ul style="list-style-type: none"> • AT_LEAST_256_BITS • EXPLICIT_CIPHERS - If this cipher class is specified, a list of ciphers must be provided in the Explicit Cipher List field. • FIPS_CIPHERS • MORE_THAN_128_BITS • NO_EXPORTABLE_CIPHERS <p>Default: AT_LEAST_256_BITS</p>
Explicit Cipher List	<p>A comma-separated list of explicitly named ciphers. For example: 'TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_RSA_WITH_AES_256_CBC_SHA256, TLS_DHE_DSS_WITH_AES_256_CBC_SHA'</p> <p>This must be provided if "EXPLICIT CIPHERS" is specified in the SSL Cipher Class field.</p>

5. Click **Save**.

Deleting an SSL Client Provider Shared Resource

An existing SSL Client Provider shared resource can be deleted using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **SSL Client Provider**.
3. From the **Shared Resources** list, select the SSL Client Provider shared resource you want to delete.
4. Click the overflow icon  , then select **Delete**.

SMTP Connection Shared Resources

An SMTP shared resource represents a connection to an SMTP (Simple Mail Transfer Protocol) server. SMTP shared resources are used by component implementations to send and receive messages to and from SMTP mail servers.

SMTP Connection shared resources can be created, edited, or deleted using the TIBCO BPM Enterprise Administrator.

Creating or Editing an SMTP Connection Shared Resource

SMTP Connection shared resources can be created or edited using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **SMTP Connection**.
3. Click .
4. Configure the SMTP Connection shared resource using the following descriptions.

Definition

Property	Description
Name	(Required) The unique name of the SMTP Connection shared resource. The name is case insensitive.
Description	A description of the SMTP Connection shared resource.
Machine Name	(Required) The name of the host that accepts incoming requests.
Port	(Required) The port number on which to listen for SMTP requests.
Timeout	The length of time in milliseconds to wait for a response from the server. A timeout of zero is interpreted as an infinite timeout.

Login credential

Property	Description
Username	The username used to authenticate to the SMTP server.
Password	The password used to authenticate to the SMTP server.

Enable SSL

Property	Description
Enable SSL	Select this option to enable SSL for the SMTP connection. If set, an SSL Client Provider name must be specified in the SSL client provider name field.
SSL client provider name	The name of an SSL Client Provider shared resource. See SSL Client Provider Shared Resources .

5. Click **Save**.

After saving the SMTP Connection Shared Resource configuration, you can return to the **Definition** tab and click **Test connection** to test the SMTP connection. On the SMTP test connection dialog, click **Test** to test the connection. You can also optionally enter an email address to have a confirmation email sent to the specified address.

Deleting an SMTP Connection Shared Resource

An existing SMTP Connection shared resource can be deleted using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **SMTP Connection**.
3. From the **Shared Resources** list, select the SMTP Connection shared resource you want to delete.

4. Click the overflow icon , then select **Delete**.

SAML Authentication Shared Resources

A SAML Authentication shared resource specifies connection details for SAML Web Profile authentication, which allows users of your application to log in using a user name and password issued by an Identity Provider (IdP) that supports SAML Web Profile.

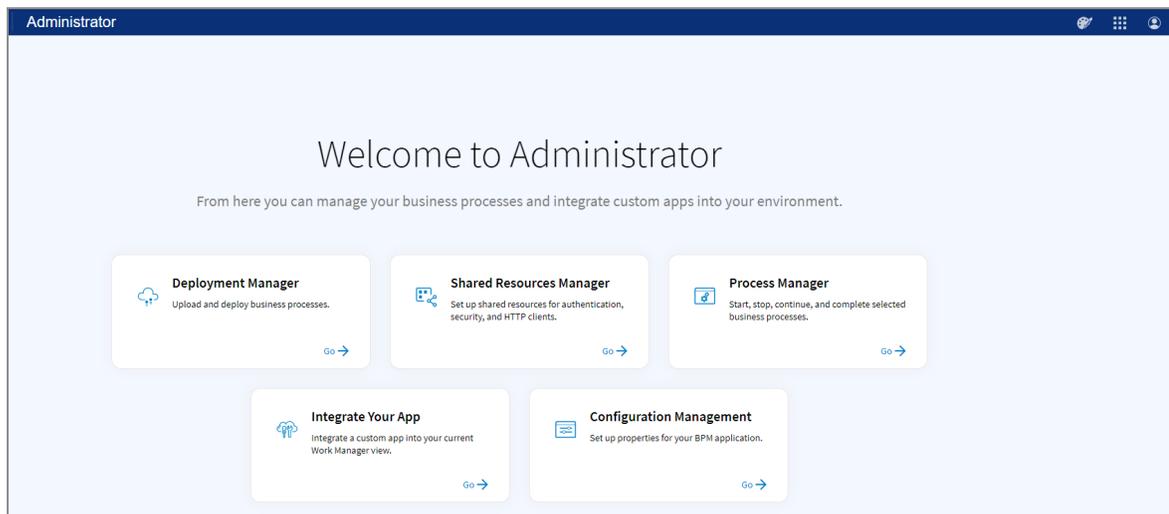
SAML Authentication shared resources can be created, edited, or deleted using the TIBCO BPM Enterprise Administrator.

Creating or Editing a SAML Authentication Shared Resource

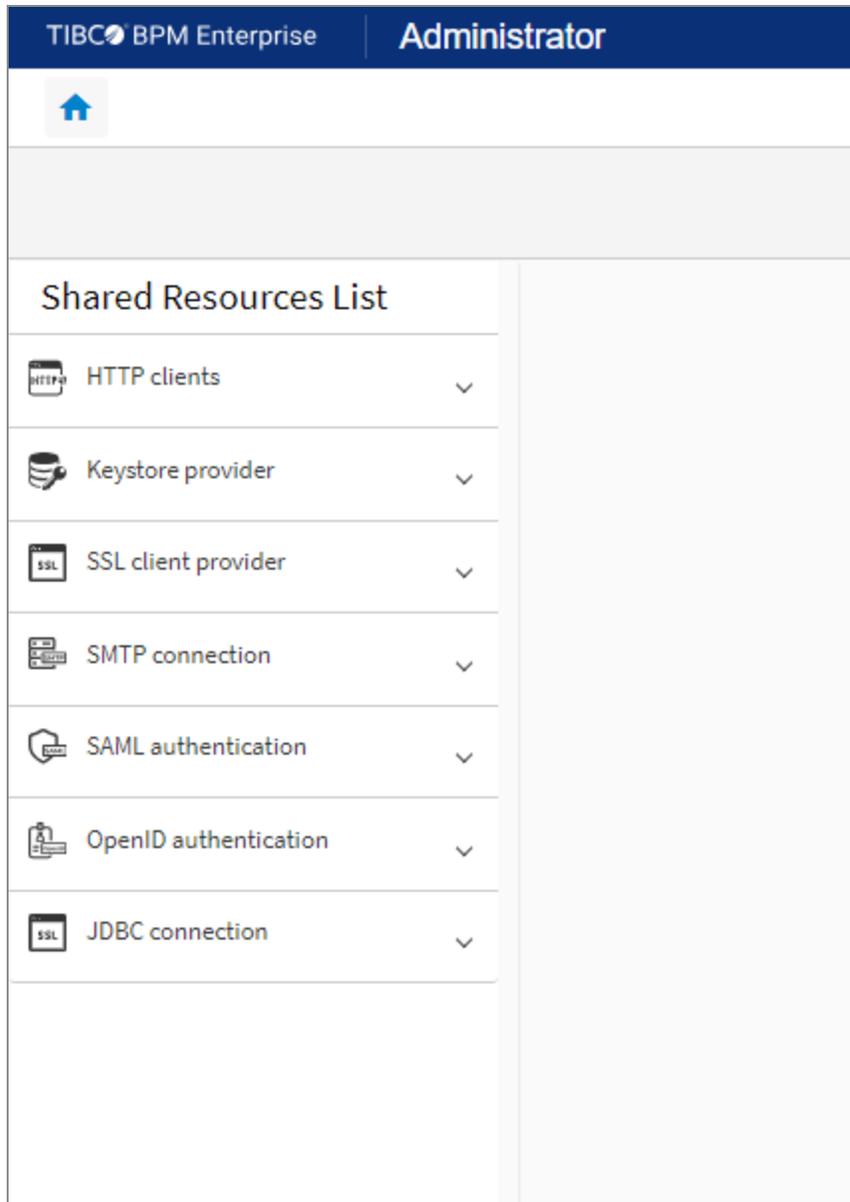
SAML Authentication shared resources can be created or edited using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise **Administrator** screen, select **Shared Resources Manager**.



2. From the **Shared Resources List** in the left pane, select **SAML Authentication**.



3. In **SAML authentication**, click .
4. Add the new SAML Authentication shared resource details based on the following description:

Definition

Property	Description
Name	(Required) The unique name of the SAML authentication shared resource. The values in this field are not case sensitive.
Description	A description of the SAML authentication shared resource.
Entity ID	(Required) A unique ID that identifies the service provider and application that has been registered with the IdP. This must match the ID that was configured at the IdP.
Response skew time	Specifies, in seconds, the maximum difference allowed between the clocks of the IdP and the TIBCO BPM Enterprise server. Default: 60
Max authentication age	Specifies, in seconds, the maximum time an authentication will remain valid. Default: 5400

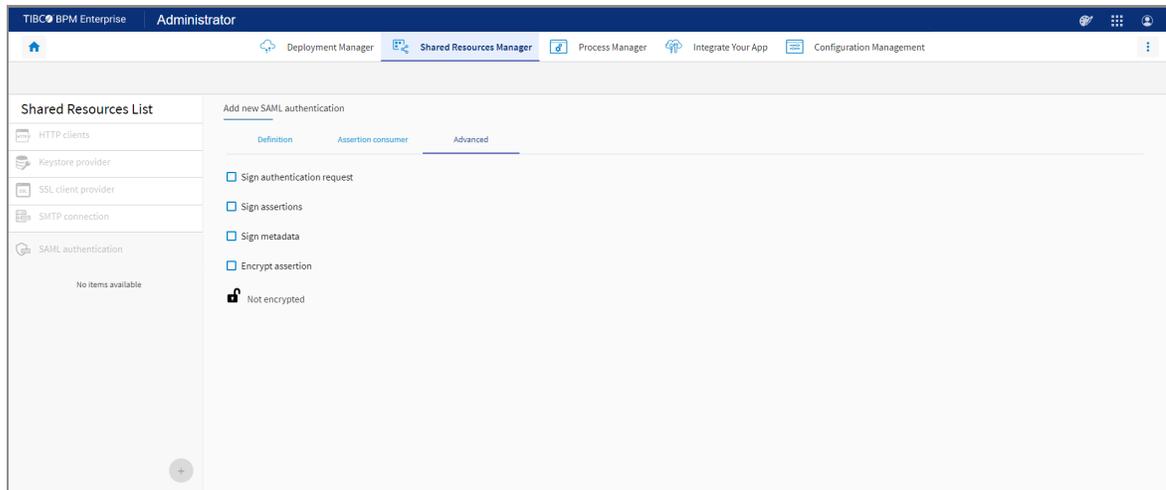
Property	Description
IdP metadata source	<p data-bbox="573 310 1395 516">Note: 'Max authentication age' value should match with 'Session duration' set on the IDP. For example, 'max Authentication Age' should have a larger value than what a user sets for 'Session never expire' in the Google IDP while configuring the SAML shared resource (for example, 30 days means 2592000 seconds).</p> <p data-bbox="553 585 1382 653">(Required) Specifies the source of the metadata file from the IdP. Click one of the following two options:</p> <ul data-bbox="602 688 1414 932" style="list-style-type: none"> <li data-bbox="602 688 1414 831">• IdP HTTP Metadata URL, if you have the URL location for the IdP metadata. Enter the HTTP URL for the location where the metadata provided by the IDP is hosted. For example, <code>http://hostname/Metadata.xml</code> <li data-bbox="602 863 1403 932">• IdP String Metadata, if you have the metadata file from the IdP. Enter the Metadata provided by the IdP. <p data-bbox="634 963 1403 1073">Alternatively, click Upload XML metadata file, if you have a local IdP metadata file. When the IdP metadata file is uploaded, the file content is populated in the text box.</p>
Entry Point URL	(Required) The relative entry point URL to initiate the SAML handshake with IdP. For example, <code>/saml/login</code>
Enabled	<p data-bbox="553 1245 1365 1352">Select this check box to enable the SAML Authentication shared resource for Single Sign-On (SSO). Currently, only one SAML Authentication shared resource can be enabled.</p> <p data-bbox="573 1398 1357 1465">Note: At any point, only a single SSO related shared resource can be enabled, that is, either SAML or OpenID.</p>

Assertion consumer

All the fields in the **Assertion consumer** tab except the **Assertion Consumer Service URL** are read-only. The values of the read-only fields are derived from the value in the **Assertion Consumer Service URL** field.

Property	Description
Assertion Consumer Service URL	(Required) Assertion Consumer Service (ACS) URL defines the endpoint that receives authentication assertions from the IdP. The IdP must use the same URL to configure Assertion Consumer Service (ACS) endpoint. For example, <code>http://bpm-host:8228/saml/SSO</code>
Base URL	The base URL where TIBCO BPM Enterprise is hosted. For example, <code>http://bpm-host</code>
Scheme	The HTTP scheme. For example, <code>http</code> or <code>https</code> . This is a part of the ACS URL.
Server name	The server name. This is a part of the ACS URL.
Server port	The server port. This is either a part of the ACS URL or takes the default value.
Context path	The context path. This is a part of the ACS URL.

Advanced



Property	Description
Sign authentication request	Select the check box to sign the authentication request. Ensure that the authentication request from the service provider is signed before you select this check box. Provide a valid public key or certificate to the IdP to identify signed requests.
Sign assertions	Select to sign assertions.
Sign metadata	Select to sign metadata.
Encrypt assertion	Select to encrypt assertion.
The following fields are displayed if any one of the above options is selected.	
Keystore provider name	(Required) The name of the KeyStore Provider used for encrypting and signing.
Key alias to encrypt	(Required when Encrypt assertion is selected) The alias of the key used for encrypting.
Key alias to encrypt	(Required when Encrypt assertion is selected) The password for the key that is used for encrypting.

Property	Description
password	
Default key alias	(Required) The alias of the default key.
Default key alias password	(Required) The password for the default key.
Key alias to sign	(Required when one of the signing related options is selected)The alias of the key used for signing.
Key alias to sign password	(Required when one of the signing related options is selected) The password for the key is used for signing.

5. Click **Save**.



Note: It is necessary to restart the TIBCO BPM Enterprise containers so the SAML shared resource can be used for authentication.

Deleting a SAML Authentication Shared Resource

An existing SAML Authentication shared resource can be deleted using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the dropdown list in the upper left, select **SAML Authentication**.
3. From the **Shared Resources** list, select the SAML Authentication shared resource you want to delete.
4. Click , then select **Delete**.

OpenID Authentication Shared Resources

OpenID Authentication shared resource specifies connection details for OpenID Connect authentication for both inbound and outbound calls.

Inbound

Provides an endpoint for the users of your application to log in to the BPME server using a username and password issued by an Identity Provider (IdP) that supports OpenID Connect. By using this type of authentication, BPME resources and services are protected for the third-party clients.

Outbound

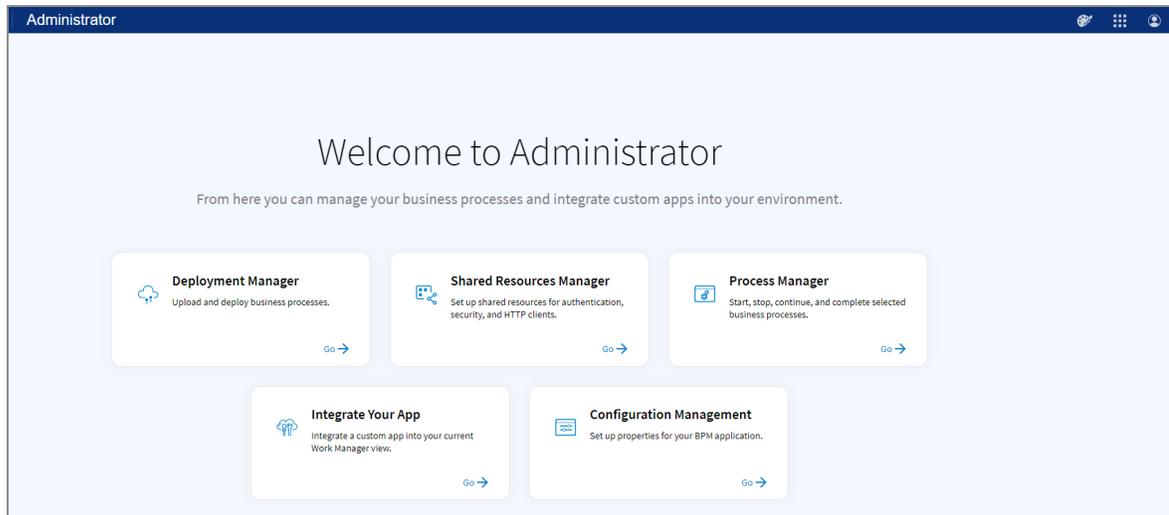
Authorizes access to the third-party client applications by using an Identity Provider (IdP) that supports OpenID Connect. You can use an outbound call from a REST service activity to invoke an external API on a third-party server from your BPME server.

Creating or Editing an OpenID Authentication Shared Resource

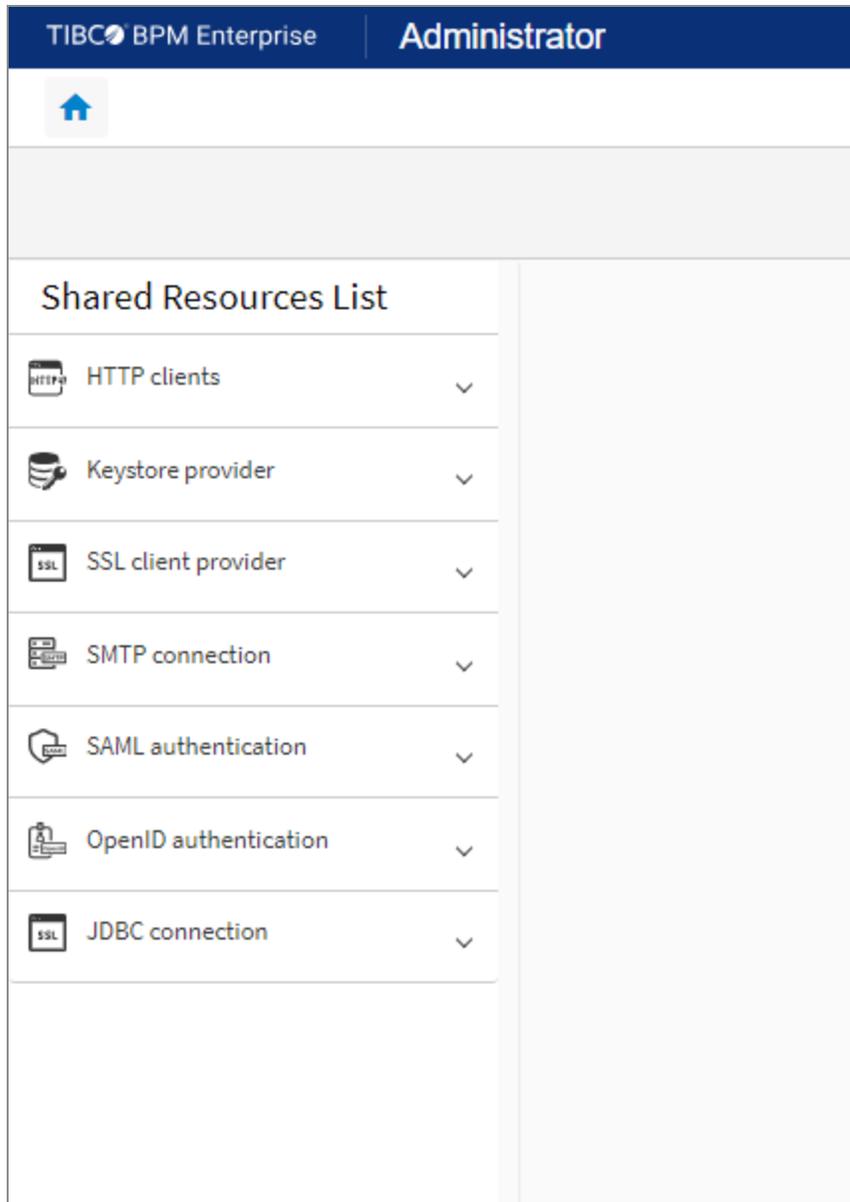
OpenID Authentication shared resources can be created or edited using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise **Administrator** screen, select **Shared Resources Manager**.



2. From the **Shared Resources List** in the left pane, select **OpenID authentication**.



3. Expand **OpenID authentication** menu and click  to add a new shared resource, or select an existing shared resource to edit it.
4. Configure the OpenID Authentication shared resource using the following descriptions.

Definition

Property		Description
Name		(Required) The unique name of the OpenID Authentication shared resource. The name value is case insensitive.
Description		A description of the OpenID Authentication shared resource.
Grant Type		<p>Defines the type of the token flow request.</p> <p>TIBCO BPM Enterprise supports the following grant types:</p> <ul style="list-style-type: none"> • Authorization Code: This grant type includes the exchange of an authorization code to authenticate API access. Authorization code is for browser (user) SSO access to TIBCO BPM Enterprise. • Client Credentials: This grant type includes the exchange of application credentials, such as client ID and client secret, to authenticate API access. Client credentials is for TIBCO BPM Enterprise to access third party services from a REST service task in a process using SSO.
Authorization Code	Client ID	(Required) The ID that identifies the client at the Identity Provider (IdP). This and Client Secret are obtained from the IdP when the client registers an application with the IdP to provide authentication for users.
	Client secret	(Required) The password for the Client ID account.
	User Key	(Optional) Specifies the claim that is used to

Property	Description
	<p>identify the user token from the list of claims that are returned from the IdP (based on the Auth scope). By default, User Key is the user's email address.</p>
Auth Scope	<p>(Read-Only) Defines the claims to be returned by the IdP when the IdP authenticates the user and issues an ID Token. These claims are user attributes that provide the application with the user details.</p> <p>Auth scope <i>openid email</i> is supported.</p>
Enabled	<p>Select this checkbox to enable the OpenID Authentication shared resource for Single Sign-On use. Currently, only one OpenID Authentication shared resource can be enabled.</p> <p>Note: At any point, only the SSO related shared resource can be enabled, that is, either SAML or OpenID.</p>
Client Credentials	<p>Invocation Type</p> <p>Select the following type of invocation from the dropdown list:</p> <ul style="list-style-type: none"> • Inbound: To access TIBCO BPM Enterprise APIs from third-party server. <p>Note: Inbound invocation type is not supported for this release.</p> <ul style="list-style-type: none"> • Outbound: To access third-party APIs from TIBCO BPM Enterprise server.

Property	Description
Client ID	(Required) The ID that identifies the client at the Identify Provider (IdP). This, and the Client Secret (see below), are obtained from the IdP when the client registers an application with the IdP to provide authentication for users.
Client Secret	(Required) The password for the Client ID account.

Shared Resources List

-  HTTP clients ▼
-  Keystore provider ▼
-  SSL client provider ▼
-  SMTP connection ▼
-  SAML authentication ▼
-  OpenID authentication ▼
-  JDBC connection ▼

Add new OpenID authentication

Definition
URI

Name *

Description

Grant Type *
Authorization Code

Client ID *

Client secret *

User key

Auth scope
openid email

Enabled

Shared Resources List	Add new OpenID authentication
HTTP HTTP clients	Definition URI
Keystore provider	Name *
SSL SSL client provider	Description
SMTP SMTP connection	Grant Type *
SAML SAML authentication	Client Credentials
OpenID OpenID authentication	Invocation Type *
JDBC JDBC connection	Outbound
	Client ID *
	Client secret *

URI

Property	Description
Access token URI	(Required) The REST OpenID token service URI, which is provided by the IdP. It is used to obtain an ID Token for the authenticated user.
Redirect URI	<p>(Required) The URI to which the IdP will send the ID Token after authentication.</p> <p>The Redirect URI appends the BPME server base URL with the user-defined configured path.</p> <p>So, if the BPME server base URL is <i>https://localhost:8443</i>, then the user can configure the redirect uri as <i>https://localhost:8443/google-login</i>, where <i>google-login</i> is the configured path. The system then dynamically hosts <i>https://localhost:8443/google-login</i> as the endpoint.</p> <p>Ensure that the path is not hierarchical, such as <i>googlelogin/subpath1/subpath2</i>.</p>
Authorization URI	(Required) The REST OpenID user claims/information service URI, which is used to obtain user profile information.

Property	Description
JSON web keyset URI	(Required) The URI to the JSON Web Key Set (JWKS), which is a JSON data structure that represents a set of public keys used to verify the signature of the JSON Web Token (JWT) issued by the IdP.

The screenshot displays the TIBCO BPM Enterprise Administrator interface. The top navigation bar includes the logo and the text 'TIBCO BPM Enterprise' and 'Administrator'. Below the navigation bar is a home icon on the left and a cloud icon on the right. The main content area is split into two columns. The left column, titled 'Shared Resources List', contains a vertical menu with the following items: HTTP clients, Keystore provider, SSL client provider, SMTP connection, SAML authentication, and OpenID authentication. The right column, titled 'Add new OpenID authentication', features two tabs: 'Definition' (selected) and 'URI'. Under the 'Definition' tab, there are five input fields, each with an asterisk indicating it is required: 'Access token URI *', 'Redirect URI *', 'Authorization URI *', and 'JSON web keyset URI *'. At the bottom of this section, there is a lock icon and the text 'Not encrypted'.

The screenshot shows the TIBCO BPM Enterprise Administrator interface. The top navigation bar includes the TIBCO logo and the text 'TIBCO BPM Enterprise' and 'Administrator'. Below the navigation bar is a home icon on the left and a cloud icon on the right. The main content area is divided into two sections. On the left is the 'Shared Resources List' with a table of resources:

Icon	Resource Name
	HTTP clients
	Keystore provider
	SSL client provider
	SMTP connection
	SAML authentication
	OpenID authentication

On the right is the 'Add new OpenID authentication' configuration page. It has two tabs: 'Definition' (selected) and 'URI'. Below the tabs is a form with a label 'Access token URI *' and a dropdown menu currently set to 'Not encrypted'.

5. Click **Save**.

i Note: You do not need to restart OpenID shared resource in the case of the OpenID client credentials outbound flow.

Deleting an OpenID Authentication Shared Resource

An existing OpenID Authentication shared resource can be deleted using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, select **Shared Resources Manager**.
2. From the list in the left pane, select **Open ID Authentication**.
3. From the list of OpenID Authentications, select the OpenID Authentication shared resource you want to delete.
4. Click , then select **Delete**.

JDBC Connection Shared Resources

The JDBC connection shared resource represents a JDBC connection that is used by database activities in TIBCO BPM Enterprise applications to access databases. JDBC connection shared resources can be created, edited, or deleted using the TIBCO BPM Enterprise Administrator.

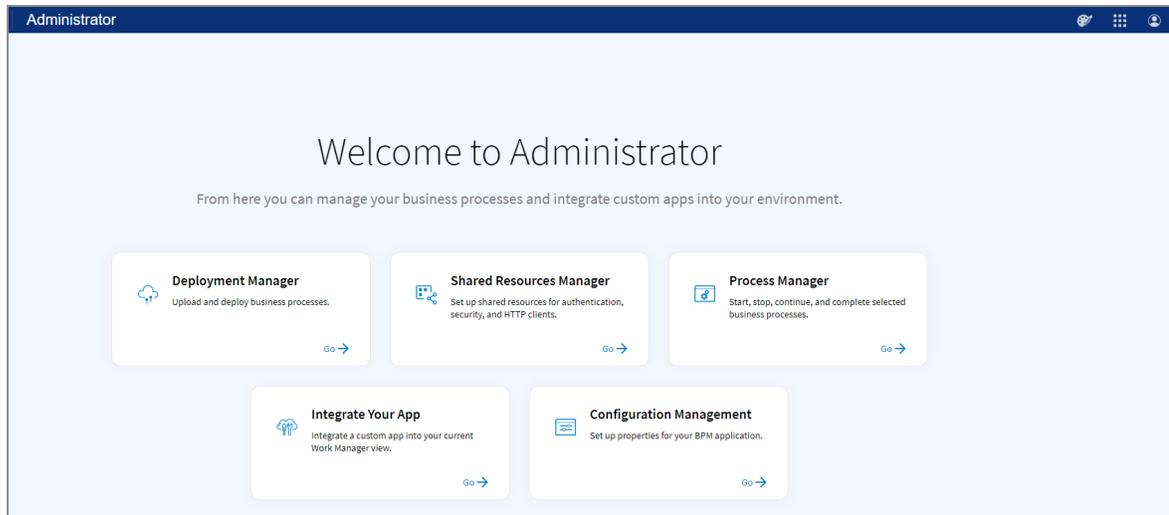
 **Note:** The JDBC connection shared resources can be configured to use either PostgreSQL or the TIBCO BPM Enterprise database type. For example, if you choose Oracle database for TIBCO BPM Enterprise during installation, you can configure a JDBC shared resource to use either an Oracle or a PostgreSQL.

Creating or Editing a JDBC Connection Shared Resource

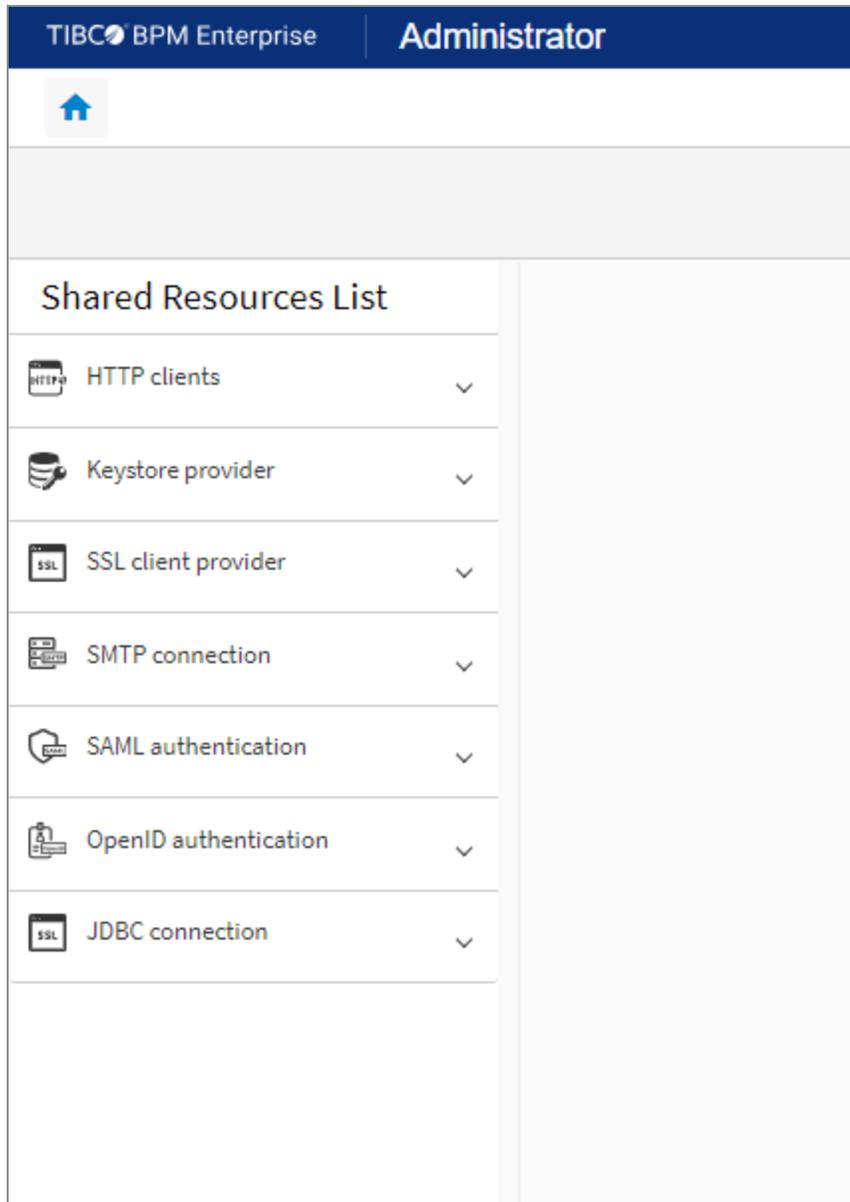
JDBC connection shared resources can be created or edited using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise **Administrator** screen, select **Shared Resources Manager**.



2. From the **Shared Resources List** in the left pane, select **JDBC connection**.



3. Expand **JDBC connection** menu and click  to add a new shared resource, or select an existing shared resource to edit it.
4. Configure the JDBC connection shared resource using the following descriptions:

Connection details

Property	Description
General	
Name	(Required) The unique name of the JDBC connection shared resource.
Description	A description of the JDBC connection shared resource.
Connection URL	(Required) The URL required to connect to the database. A template of the URL is supplied for the driver you select in the Driver Class field or you can type the URL: <pre>jdbc:postgresql://<servername>:<portnumber>/<dbname></pre>
Username	(Required) The username used to authenticate connections to the server.
Password	(Required) The password used to authenticate connections to the server.
Driver Class	(Required) The name of the JDBC driver class. You can select from a dropdown menu of supported drivers.
SSL	
Enable SSL	Enable SSL connections. When the check box is selected, the SSL Client Provider Name field is displayed. Default: Not selected.
SSL Client Provider Name	(Required if Enable SSL is selected.) The name of an SSL Client Provider. For more details, see SSL Client Provider Shared Resources .

TIBCO BPM Enterprise | Administrator

[Home](#)

Shared Resources List

-  HTTP clients ▼
-  Keystore provider ▼
-  SSL client provider ▼
-  SMTP connection ▼
-  SAML authentication ▼
-  OpenID authentication ▼
-  JDBC connection ▼

Add new JDBC connection

[Connection details](#)

General

Name *

Description

Connection URL *

Username *

Password *

Driver Class * ▼

SSL

Enable SSL

Advanced

Property	Description
Connection pool and cache	
Pool initial value	The initial number of connections maintained in the pool.
Pool min value	The minimum number of connections in the pool.
Pool max value	The maximum number of connections in the pool.
Statement cache	The size of the prepared statement cache in the connection pool that is created. Default: 0 (No statement cache)
Timeout	
Read timeout	The number of milliseconds required to receive the response from the server. Default: 60000
Connection Timeout	The number of milliseconds to wait for a connection to be established before timing out. Default: 60000

The screenshot displays the TIBCO BPM Enterprise Administrator interface. The top navigation bar includes the TIBCO BPM Enterprise logo and the title 'Administrator'. A 'Deployment Manager' link is visible in the top right corner. The main content area is split into two columns. The left column, titled 'Shared Resources List', contains a list of shared resources with expandable dropdown menus: HTTP clients, Keystore provider, SSL client provider, SMTP connection, SAML authentication, OpenID authentication, and JDBC connection. The right column, titled 'Add new JDBC connection', features two tabs: 'Connection details' (selected) and 'Advanced'. Under the 'Connection details' tab, there are several configuration sections: 'Connection pool and cache' with fields for 'Pool initial value', 'Pool min value', and 'Pool max value'; 'Statement cache'; and 'Timeout' with fields for 'Read timeout' and 'Connection time...'. The 'JDBC connection' resource in the list is highlighted with a blue background.

Referencing applications

A list of applications using the JDBC connection shared resource is displayed.

5. Click **Save**.

Deleting a JDBC Connection Shared Resource

An existing JDBC connection shared resource can be deleted using the TIBCO BPM Enterprise Administrator.

Procedure

1. From the TIBCO BPM Enterprise Administrator, go to **Shared Resources Manager**.
2. From the list in the left pane, expand **JDBC connection**.
3. From the list of JDBC connections, select the JDBC connection shared resource you want to delete.
4. Click the overflow icon , then click **Delete**.

Organization Browser

The Organization Browser can be used to browse organization models, create LDAP containers that hold potential resources, map resources to groups and positions in the organization model, and edit various organizational entity information.

Organization models are created using TIBCO Business Studio - BPM Edition. For information about organization model creation, see "Organization Model Creation" in the *TIBCO Business Studio - BPM Edition Application Designer's Guide*.

Accessing the Organization Browser

To access the organization browser, a user should be in the System Administrator group in Organization Model Version 0 or should have the system administration, organization admin, LDAP admin, and resource admin system action.

Procedure

1. Enter the following URL in your browser:

```
protocol://host:port/apps/login
```

where:

- *protocol* is the communications protocol being used, either http or https. This was specified at installation.
- *host* is the DNS name or IP address of the server hosting the TIBCO BPM Enterprise runtime.

- *port* is the port being used. The default value is 80.
2. Log in with a valid TIBCO BPM Enterprise username and password.
 3. Click .
 4. Click **Organization Browser**.

Browsing the Organization Model

The Organization Browser can be used to browse the organization model, which can consist of organizations, organization units, positions, and groups.

Organization Model Versions

As your organization changes over time, your organization model may need to be modified to add or remove groups, positions, organization units, and so on. Additions to the organization model can be merged into the existing version, but when parts of the model are deleted or other types of significant changes are made, it is given a new version number.

When you are browsing the organization model, mapping resources, or performing any function available in the Organization Browser, you must select the version of the organization model that you want to be working with. This can be done using the **Version** field in which you can select the version of the organization model you want to work with. For example:



As you select different versions, the contents of that version of the organization model are displayed.

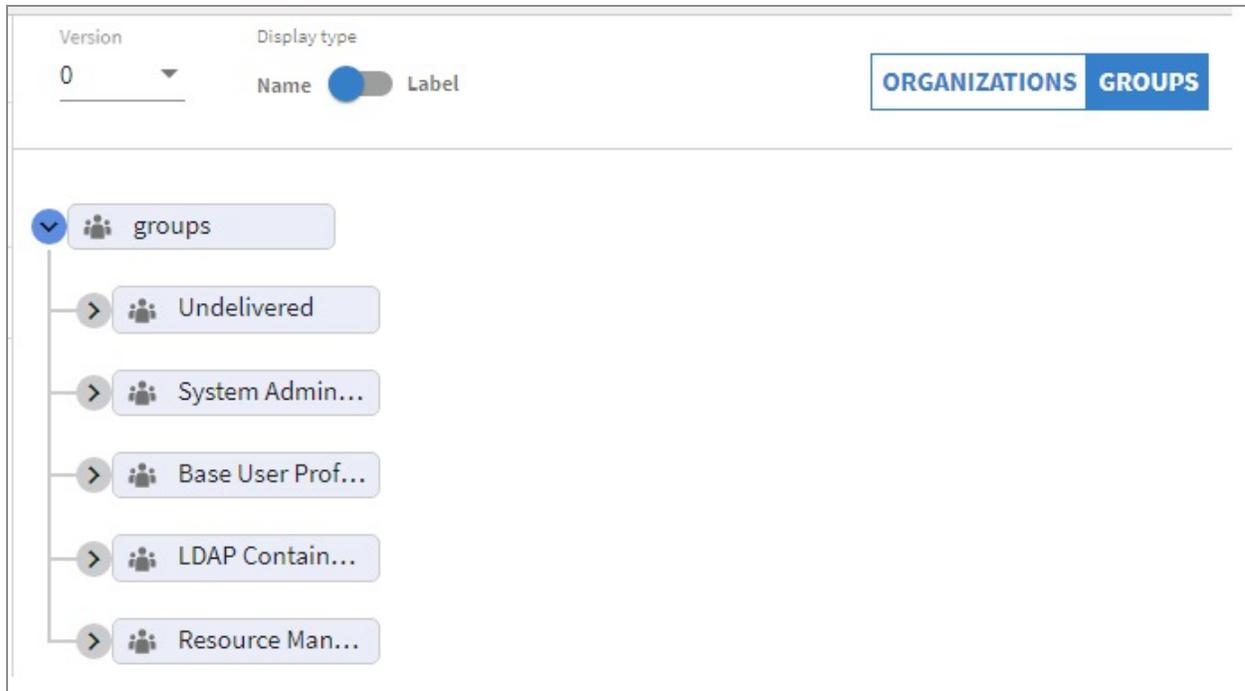
You may need to select an earlier version of the organization model to map resources to groups and/or positions in that version because processes being run may still use that earlier version.

Note that the version number shown in the **Version** field is the *major* version number.

Organization Model Version 0

Version 0 of the organization model is built into the system by default.

The following illustrates the default groups in Organization Model Version 0 (it does not contain any organization units or positions):



Note that after you select Version 0 in the **Version** field drop-down, you must click **GROUPS**. By default, **ORGANIZATIONS** is automatically selected when you select a different organization model version.

The System Administrator and Undelivered groups in Version 0 of the organization model are described below:

- **System Administrator** - Users that are mapped to this group have access to all functions in BPM applications. By default, there is a "System Administrator user" who is a member of this group and cannot be deleted from it. You can map additional users to this group, which also gives them access to all functions in the BPM application.

i Note: Out-of-the-box, the System Administrator user's user name and password are “tibco-admin” and “secret”, respectively, but may have been changed.

- **Undelivered** - This is a special group to which work items are sent that for some reason could not be delivered to the intended user. The tibco-admin user is a member of this group, and cannot be removed from it. You can add additional users to this group, if desired. Note, however, that you cannot distinguish undeliverable work items from work items that would be received because of membership in other groups or positions, so only a user who would deal with those types of work items should be mapped to this group.

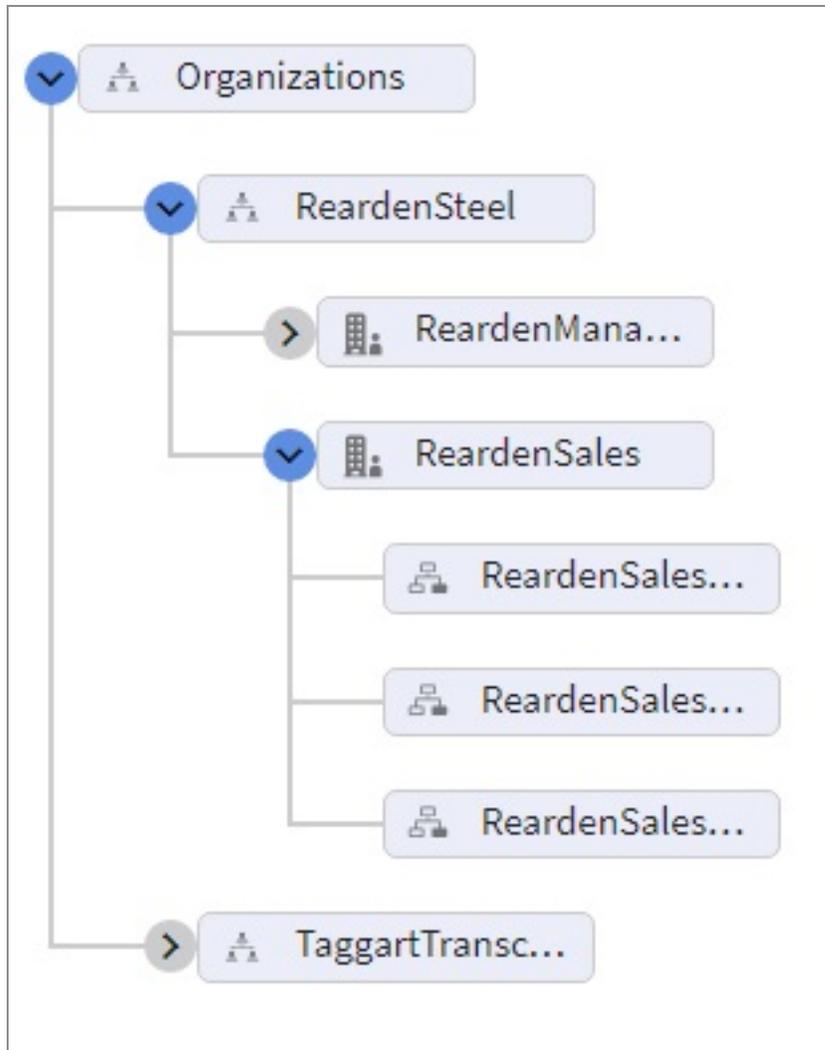
i Note: The Base User Profile, LDAP Container Managers, and Resource Managers groups are not used at this time.

Browsing Organization Units and Positions

An organization model can consist of multiple organization units, which can each contain multiple positions.

Procedure

1. Access the Organization Browser and click [Browse Organization](#).
2. Click **ORGANIZATIONS**.
3. Ensure that the appropriate version of the organization model is selected in the **Version** field. A graphical representation of the organization units, and their respective positions, that have been defined in the selected organization model version is displayed. Expand the model to see the defined organization units and positions. For example:

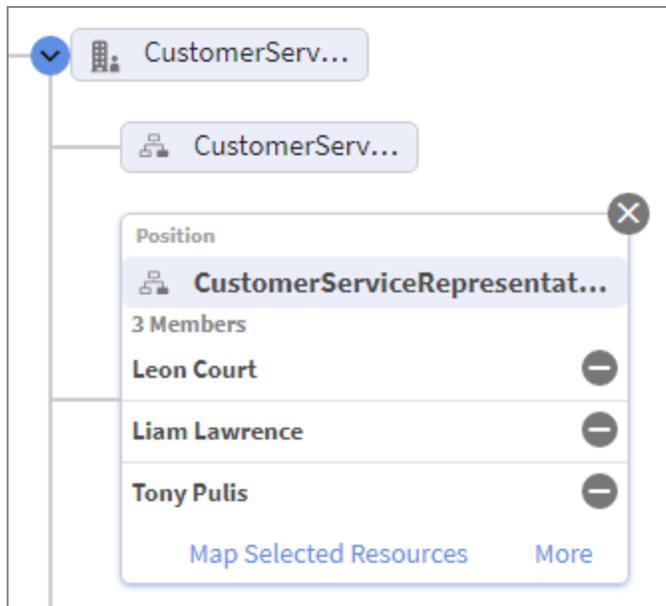


The icon to the left of each organizational entity indicates the type of entity, as follows:

Icon	Entity Type
	Organization
	Organization Unit
	Position

i Note: If LDAP containers have been created on your system that have an organization relationship set up, you may or may not be able to see all organizations in the organization model when you display the Organization Browser. For more information, see [Container Organization Relationships](#).

4. Determine how many members are in a particular position by clicking on the position in the organization model. For example:



For information about adding new members to a position, see [Mapping Resources](#).

5. View additional information about a selected organization unit or position by clicking the **More** link.

A display similar to the following is shown:

 CustomerServiceRepresentative
✕

Parent Group

CustomerService

 **Members (3)** ▼

 **Privileges (1)** ▼

 **Required Capabilities (3)** ▼

 **Push Destinations (0)** ▼

 **Candidate Query (0)** ▼

Each of these items can be expanded by clicking the ▼ character to the right of the item. The following types of details are provided:

Property	Description
Members	The number of resources that are mapped to the selected position. This is only applicable to positions, as resources cannot be mapped to organization units.
Privileges	The privileges assigned to an organizational unit or position, which are inherited by resources mapped to the position. For more information, see Privileges .
Required Capabilities	The capabilities that resources should possess to be mapped to the position. Note, however, that this is not an enforced requirement — the Organization Browser will allow you to assign a resource that does not have the required capabilities to a group or position that has been assigned that capability.

Property	Description
	<p>This is only applicable to positions, as capabilities cannot be assigned to organization units.</p> <p>For more information, see Capabilities.</p>
Push Destinations	<p>The destination(s) to which work items sent to the organizational entity are to be pushed. For more information, see Push Destinations.</p>
Candidate Query	<p>An LDAP query that determines how a position or group is populated. For information, see Candidate Queries.</p>

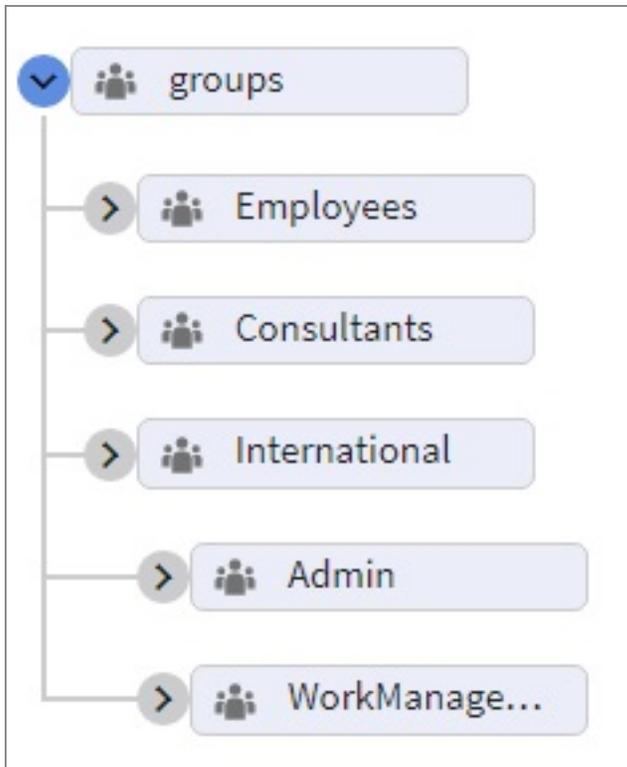
Browsing Groups

An organization model can consist of multiple groups, including subordinate groups.

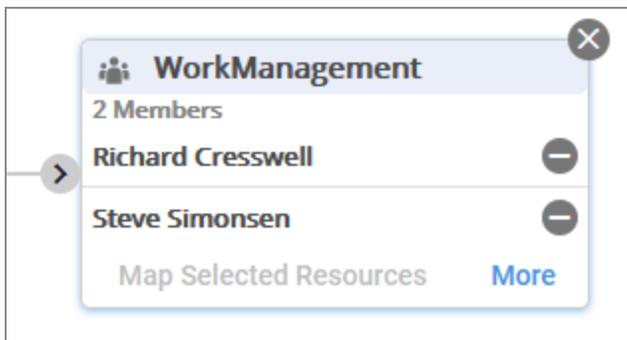
Procedure

1. Access the Organization Browser and click [Browse Organization](#).
2. Click **GROUPS**.
3. Ensure that the appropriate version of the organization model is selected in the **Version** field.

A graphical representation of the groups that have been defined in the organization model is displayed. For example:



- Determine how many members are in a particular group by clicking on the group in the organization model. For example:



For information about adding new members to a group, see [Mapping Resources](#).

- View additional information about a selected group by clicking the **More** link. A display similar to the following is shown:

WorkItemManagement ✕	
Parent Group	GlobalWorkManagement
 Members (2)	▼
 Privileges (0)	▼
 Required Capabilities (0)	▼
 Push Destinations (0)	▼
 Candidate Query (0)	▼
 Miscellaneous properties (1)	▼

Each of these items can be expanded by clicking the ▼ character to the right of the item. The following types of details are provided:

Property	Description
Members	The number of resources that are mapped to the selected group.
Privileges	<p>The privileges assigned to the group, which are inherited by resources mapped to the group.</p> <p>For more information, see Privileges.</p>
Required Capabilities	<p>The capabilities that resources should possess to be mapped to the group. Note, however, that this is not an enforced requirement — the Organization Browser will allow you to assign a resource that does not have the required capabilities to a group or position that has been assigned that capability.</p> <p>For more information, see Capabilities.</p>

Property	Description
Push Destinations	The destination(s) to which work items sent to the organizational entity are to be pushed. For more information, see Push Destinations .
Candidate Query	An LDAP query that determines how a position or group is populated. For information, see Candidate Queries .
Miscellaneous properties	This lists various properties that have been defined for the group.

LDAP Containers

LDAP (Lightweight Directory Access Protocol) is an application protocol for querying and modifying directory services. LDAP containers are associated with an LDAP source. An LDAP source represents an LDAP server, which holds information about candidate resources — users — who may need to use or participate in BPM applications.

You must create at least one LDAP container from which resources can be selected and mapped to groups or positions in the organization model. You can create additional LDAP containers, if desired. The additional containers may contain different LDAP sources, or they may query the same LDAP sources in a different way, resulting in a different set of resources to choose from.

i Note: The recommended best practice is to create LDAP containers that show only a constrained view of the corporate LDAP. That view would ideally include only those resources that have a business role in common, that belong to a particular department, work on a particular project, etc.

The resources in an LDAP container are considered *candidate resources*, that is, resources that are available to map to groups and positions in the organization model.

When you create an LDAP container, you can use either an *LDAP query source* or an *LDAP group source* to identify the candidate resources in the LDAP directory, as follows:

- Using an LDAP Query Source - An LDAP query is used to identify the directory entries that will be candidate resources. For more information, see [LDAP Query Sources](#).

- Using an LDAP Group Source - A group DN (distinguished name) is used to identify the LDAP directory that is the group. When a group DN is specified, a member attribute is also specified, which holds the collection of member identifiers, that is, their DNs. This provides the list of candidate resources. For more information, see [LDAP Group Sources](#).

Primary and Secondary Sources

Every LDAP container must include one *primary* LDAP source. It can also include one or more *secondary* LDAP sources.

The primary LDAP source identifies the candidate resources that are available to map to groups and positions in the organization models.

If there are secondary LDAP sources defined, they are used to find additional information about each candidate resource.

Lookups are performed into each secondary LDAP source. If an exact match of a candidate resource can be found in every secondary LDAP source, the data from all sources is merged. This is accomplished using the attribute relationships you specify when adding a secondary LDAP source to a container.

The following are reasons you might want to define a secondary LDAP source:

- The business process needs to access attribute data that is in both the primary and secondary LDAP sources.
- The business process needs to access attribute data from an LDAP source that is not used for login authentication (the primary LDAP source is always used for authentication).

The Organization Browser constructs the list of candidate resources as follows:

- It starts with a list of candidate resources from the primary LDAP source.
- An attempt is made to match those candidate resources with entries in each secondary resource. If an exact match is found in every secondary LDAP source, the data from the secondary sources is merged with the data from the primary source.

If a candidate resource is not found inside one or more of the secondary LDAP sources, the candidate resource is eliminated from the list.

If matches are found in every secondary LDAP source, they must uniquely identify only one LDAP entry in each source. If one or more match multiple items, the item

remains in the candidate resources list but is marked as invalid.

As an example, suppose you have two LDAP sources: Acme-Employees and Acme-Developers. The Acme-Employees LDAP source includes sales and support resources, as well as developers. The Acme-Developers LDAP source includes Acme employees who are developers, as well as developer contractors who are not Acme employees. If you want the list of candidate resources to include all Acme employees that are developers, and the business process needs attribute data from both LDAP sources, you would add both sources to one container, using filter criteria to filter out all resources other than Acme developers.

Conversely, if the attribute data needed by the business process is available in one of the LDAP sources, it is much more efficient to include only one LDAP source in the container.

Also note that when you specify primary and secondary LDAP sources, you can use either an LDAP query or an LDAP group DN to identify the candidate resources to include in the LDAP container.

Multiple Entries

When you are defining a secondary LDAP source, you must choose attributes from both the primary and secondary source whose values are compared to determine the final set of resources to include in the LDAP container. The goal is to choose the appropriate attributes so that you know the resources used from the secondary source are the same as in the primary source. If you choose attributes where it cannot make a one-to-one match, you may see a "Multiple Entries" message next to a resource name.

This could occur, for instance, if you were comparing attributes between the primary and secondary sources that contained only the resource's last name (for example, the `sn` attribute), and multiple entries contain that last name.

LDAP Query Sources

If you are using an LDAP query source to identify the candidate resources to include in an LDAP container, a filter string is used to determine which of the resources to return from the LDAP source.

When using an LDAP query source, the following two attributes are used to determine the candidate resources for the LDAP container:

- LDAP Alias - The name (or alias) of the LDAP source from which candidate resources

will be obtained.

- LDAP Query - A filter string that will be used to determine which of the resources to return from the LDAP source. This allows you to limit the resources returned. For example, you may only be interested in considering resources from a specific department or region.

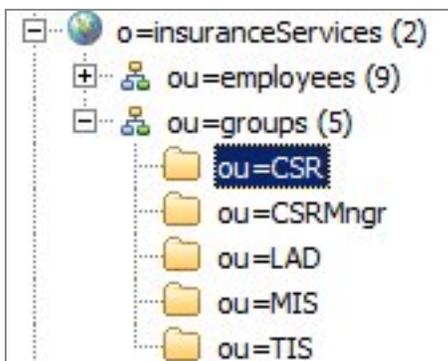
For more details about the parameters that can be specified when creating an LDAP container using an LDAP query, see [Creating an LDAP Container using an LDAP Query](#).

LDAP Group Sources

If you are using an LDAP group source to identify the candidate resources to include in the LDAP container, you will specify a group DN to identify the directory entry that is the group. When a group DN is specified, a member attribute is also specified, which holds the collection of member identifiers, that is, their DNs. This provides the list of candidate resources.

When using an LDAP group source, the following three attributes are used to determine the candidate resources for the LDAP container:

- LDAP Alias - The name (or alias) of the LDAP source from which candidate resources will be obtained.
- Group DN - The LDAP directory entry that is the group. For example, if the following LDAP directory identifies the group, the Group DN is "OU=CSR,OU=groups,O=insuranceServices":



- Member attribute - Identifies the attribute within the group entry that holds the collection of DNs that identify the candidate resources. The following example shows the attributes for the DN shown above:

DN: ou=CSR,ou=groups,o=insuranceServices	
Attribute Description	Value
objectClass	organizationalRole (structural)
objectClass	top (abstract)
cn	Customer Services Representative
ou	CSR
roleoccupant	uid=jparkin,ou=Employees,o=insuranceServices
roleoccupant	uid=rcresswell,ou=Employees,o=insuranceServices

In this example, "roleoccupant" would be the member attribute, as it provides the DN for each member of the group.

For more details about the parameters that can be specified when creating an LDAP container using an LDAP group, see [Creating an LDAP Container using an LDAP Group](#).

Object Classes

All entries in an LDAP directory are of a particular object class, that is, their "objectClass". The objectClass defines what attributes the directory entry "must" have (required attributes), as well as the attributes it "may" have (optional attributes).

LDAP directories that contain group entries, often have one of the following objectClasses:

- groupOfNames
- groupOfUniqueNames
- groupOfURLs
- organizationalRole

These common group-related object classes contain an attribute whose values identify members of the group. The member attributes for the common group-related object classes are "member", "uniqueMember", "memberURL", and "roleoccupant", respectively.

As you can see in the example above, the directory identified by the DN "OU=CSR,OU=groups,O=insuranceServices" is an objectClass of organizationalRole, and has two "roleoccupant" attributes that contain the DNs of the members of the group (CSRs in this example).

Static LDAP Groups

Static LDAP groups specify the DN of each member of the group in the member attribute. The example shown above is a static group.

Another example of a static LDAP group directory is the following:

```
DN: cn=Dev.Staff,ou=Austin,c=US
objectclass: groupOfNames
cn: Dev.Staff
member: cn=John Doe,o=IBM,c=US
member: cn=Jane Smith,o=IBM,c=US
member: cn=James Smith,o=IBM,c=US
```

In this example, the group members are identified in the "member" attribute - each attribute contains the DN of a member of the group.

Static LDAP groups can also be nested by specifying the DN of another group as a value of a member attribute. If any value within a member attribute identifies another group, the attribute of that group with the same name is used to augment the collection of group members. The resulting candidate resources consist of all nested group members.

The identification of nested groups is done using their objectClass; comparing the objectClasses named in the Directory Engine configuration (for information, see "Directory Engine Configuration" in the *TIBCO BPM Enterprise Administration* guide).

The known group objectClasses, groupOfNames, and groupOfUniqueNames, are applied by default. Those member entries that are of any of the named objectClasses are considered to be nested groups.

Dynamic LDAP Groups

Dynamic groups specify one or more URL search filters (queries). All entries that match the URL search filters are members of the group. Membership of a dynamic group is defined each time the filters are evaluated.

Dynamic groups use one of the following object classes and attribute pairs:

- The groupOfURLs object class, with the memberURL attribute.
- The groupOfUniqueNames object class, with the uniqueMember attribute.

The memberURL attribute and the uniqueMember attributes specify one or more URL search filters. An example is:

```
dn: cn=GROUP1,ou=Austin
objectclass: groupOfURLs
cn: GROUP1
memberURL: ldap:///cn=users,ou=Austin??one?(group=GROUP1)
```

Creating an LDAP Container

You must create at least one LDAP container from which resources can be selected and mapped to groups or positions in the organization model.

For each LDAP container that you create, you must specify an *LDAP source* (which equates to an LDAP server) from which resources are obtained.

The procedure you use to create an LDAP container differs, depending on whether the LDAP source is an LDAP query source or an LDAP group source. See:

- [Creating an LDAP Container using an LDAP Query](#)
- [Creating an LDAP Container using an LDAP Group](#)

Creating an LDAP Container using an LDAP Query

If you are using an LDAP query source in the LDAP container definition, all resources in the LDAP source that satisfy the LDAP query are included in the list of candidate resources in the container.

Procedure

1. Open the Organization Browser (see [Accessing the Organization Browser](#)).

If any LDAP containers had previously been defined, they are shown in the left pane of the dialog box, and the details of the selected container are shown in the right pane.

If no LDAP containers have been created, the only thing displayed is the **Create New LDAP Container** button.

i Note: After you've defined a container, you can edit it by selecting the container in the list, then clicking  **>Edit**. You can generally follow the same steps in this procedure to edit an existing LDAP container.

However, if the existing LDAP container contains resources, the LDAP source for that container is fixed; you cannot delete or change the LDAP source.

Also note that if an LDAP source defined in an LDAP container is currently offline, you cannot edit the container until the LDAP source is back online.

2. Click **Create New LDAP Container**.
3. On the New LDAP Container dialog box, enter a name and an optional description for the new container.
4. In the **Select Organizations** section, you can optionally specify *organization relationships* for the new container (note that this is applicable only if there are multiple organizations in your organization model).

If the container has a relationship with an organization, resources in the container will be able to see that organization in the Organization Browser, as well as organizations that do not have an explicit relationship with a container. Resources can be mapped to positions in organizations that the user can see in the Organization Browser.

For more information about organization relationships, see [Container Organization Relationships](#).

If you are not specifying organization relationships for this container, proceed to the next step. (You can specify organization relationships for this container at a later time.)

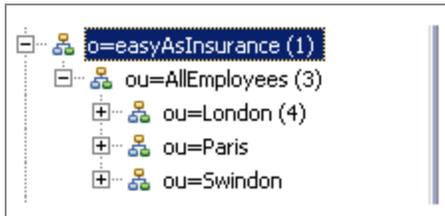
To specify organization relationships:

- a. Click the check box to the left of each organization for which the LDAP container is to have a relationship.
5. Click **Next**.
 6. Ensure that the **Query source** option is selected.
 7. In the **Alias** field, select the LDAP source from which you want to obtain resources.

The names in the **Alias** list are user-readable names that an administrator has assigned to each of the LDAP Servers available in the enterprise.

8. In the **Base dn** field, enter the branch (for example, an organization unit) in which you would like to limit the search in the LDAP directory structure. This increases the efficiency of the search if the LDAP contains a large number of branches.

The search base must provide the complete path to the desired branch in the LDAP directory structure. For example (this illustration is from an external application that shows the LDAP source), if you want to limit the search to the “London” organization unit in the following LDAP source.



You would enter the following in the **Base dn** field:

`ou=London, ou=AllEmployees, o=easyAsInsurance`

You can also leave the **Base dn** field blank, which causes the entire LDAP directory structure to be searched.

9. In the **Query** field, enter a filter string.

The filter string is used to determine which of the resources to return from the LDAP source. This allows you to limit the resources returned. For example, you may only be interested in considering resources from a specific department or region.

Query strings must be enclosed in parentheses. This allows you to specify multiple strings, each one enclosed in its own parentheses.

For information about special characters that can be used in LDAP query strings, plus some examples of query strings, see [LDAP Query String Characters and Examples](#).

10. In the **Resource name attribute(s)** field, enter one or more LDAP attributes by which you want the resources to be displayed in the list of candidate resources.

i Note: Click **Show Sample Data** to view the available resource attributes. This button must be clicked before you can advance to the next screen, so click it now to determine the resource attribute(s) that should be specified in the **Resource name attribute(s)** field.

The resource name attribute is significant for the following reasons:

- It specifies the name by which the user must log in to the TIBCO BPM Enterprise.
- It specifies the name by which the resource is listed when mapping resources to groups and/or positions in the organization model. That is, they must be names that the user doing the mapping can use to uniquely identify the resources. For example, you probably wouldn't want to use only "sn" (surname), as that may not be unique among all resources.

For example, the following shows resources when resource attributes were chosen to identify the resources by both forename and surname:

<input type="checkbox"/>	Select all resources
<input type="checkbox"/>	 Clint
<input type="checkbox"/>	 John
<input type="checkbox"/>	 Jon
<input type="checkbox"/>	 Leon
<input type="checkbox"/>	 Liam Lawrence
<input type="checkbox"/>	 Richard
<input type="checkbox"/>	 Steve
<input type="checkbox"/>	 Tony

The default resource attribute is “cn”, which typically contains a full name. But depending on the data in the LDAP source, there may be more suitable attributes for this use.

You can specify multiple attributes in the **Resource name attribute(s)** field. For instance, you could enter “givenname sn” to display the resource’s first name and last name (again, depending on what is stored in those attributes on the chosen LDAP source).

Once you save the LDAP container you will be able to view the list of resources for the container. The LDAP entities that are in this list, but have not been added, the resource name will be constructed based on the resource name attribute(s).

After a resource has been added (either using the **Add Resource** function or by mapping the user to a group or position), you can edit a resource's resource name by selecting the resource, then click .

It is possible to change the resource name attribute setting for the container, but that will not affect the resource name of resources that have already been added. It will, however, change the name that is constructed for the remaining LDAP entries that have not yet been added. For instance, using the example shown above, if you change the value in the **Resource name attribute(s)** field to “cn” (which contains "Mr" or "Mrs" with the resource's full name), the resource names now appear as shown below in the list of candidate resources.

<input type="checkbox"/>	Resources
<input type="checkbox"/>	 Leon Court
<input type="checkbox"/>	 Liam Lawrence
<input type="checkbox"/>	 Mr Jon Parkin
<input type="checkbox"/>	 Mr Richard Cresswell
<input type="checkbox"/>	 Mr Steve Simonsen
<input type="checkbox"/>	 Mr Tony Pulis

Notice that the resources that had been previously added (those that have the ) , are shown with the resource names they had when they were added; the resources that have not been added yet (blank check box), are shown with the new resource names.

11. Using the **Search scope** options, specify the depth to perform the search in the LDAP directory structure, as follows:
 - **One Level** - Only the elements directly within the base DN level are searched.
 - **Sub Tree** - Elements directly within, and below, the base DN level are searched.
12. Click **Save LDAP Source**.
13. Optionally, click **New LDAP Source**.

This optional step is used to define a secondary LDAP source. You can define one or more secondary LDAP sources. For more information, see [Primary and Secondary](#)

Sources.

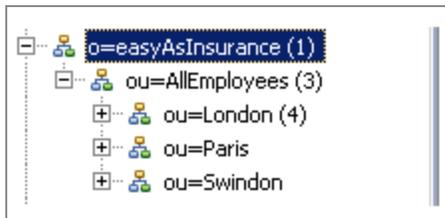
- a. Ensure that **Query source** is selected as the source type.
- b. From the **Alias** field dropdown list, select the secondary LDAP source you would like to add to the LDAP container.

Note that when choosing a secondary source, the alias that was chosen for the primary source is omitted from the **Alias** field dropdown list.

- c. In the **Base dn** field, enter the branch (e.g., an organization unit) in which you would like to limit the search in the LDAP directory structure.

This increases the efficiency of the search if the LDAP contains a large number of branches.

The search base must provide the complete path to the desired branch in the LDAP directory structure. For example, if you want to limit the search to the “London” organization unit in the following LDAP...



... you would enter the following in the **Base dn** field:

ou=London, ou=AllEmployees

- d. In the **Query** field, enter a filter string that will be used to determine which of the resources to return from the LDAP source.

The filter string is used to determine which of the resources to return from the LDAP source. This allows you to limit the resources returned. For example, you may only be interested in considering resources from a specific department or region.

Query strings must be enclosed in parentheses. This allows you to specify multiple strings, each one enclosed in its own parentheses.

For information about special characters that can be used in LDAP query strings, plus some examples of query strings, see [LDAP Query String Characters and Examples](#).

i **Note:** The **Resource name attribute(s)** field has no meaning when you are defining a secondary LDAP source.

e. Using the **Search scope** options, specify the depth to perform the search in the LDAP directory structure, as follows:

- **ONE LEVEL** - Only the elements directly within the Base-DN level are searched.
- **SUBTREE** - Elements directly within, and below, the Base-DN level are searched.

f. Click **Show Sample Data**.

This allows you to see the attributes that are in your chosen LDAP source. Notice that when you click **Show Sample Data**, two mapping fields (**primary attribute** and **secondary attribute**) appear at the bottom of the screen. These are used to choose related LDAP attributes, as described in the following steps.

g. At this point, you need to determine which attributes in the secondary LDAP source will be compared to attributes in the primary LDAP source.

The goal of comparing primary and secondary attributes is to ensure that the data from the secondary LDAP source is only merged with the appropriate candidate resource from the primary LDAP source.

Where a match cannot be found, or where it is not one-to-one, the candidate resource will not have a complete, accurate set of information, and it will be either omitted (where no match is found) or marked as "multiple entries" (where the match isn't one-to-one).

For example, if there are several resources with the same last name, you need to check more than just the last name. In some cases, checking first name will suffice, in some you may need to check more attributes (because there may be multiple resources in the secondary LDAP source with the same first and last name — and the system would not know which to include). Other types of data, such as an employee ID would work better and avoid inconsistencies in data entry (typos, nicknames, abbreviations, etc.).

You may need to go back and view the data in the attributes in the primary

LDAP source.

For example, if you know that the “ou” attribute in the primary LDAP source contains the complete name of the resources (Clint Hill, John Eustace, etc.), and the “displayname” attribute in the secondary LDAP source contains the same information, those attributes would be prime candidates to link.

i **Note:** If you choose attributes that contain names, always be aware that there may be differences in the way those names were entered in the different LDAP sources, for example, Bob vs. Robert, Mike vs. Michael, or simple misspellings. Things like employee numbers tend to make good attributes to link.

For additional information, see [Primary and Secondary Sources](#).

- h. In the **primary attribute** field, choose the primary LDAP source attribute (for example, "ou") that contains data you want to compare to the data in the attribute you will choose in the next sub-step.
- i. In the **secondary attribute** field, choose the secondary LDAP source attribute (for example, "displayname") that contains data you want to compare to the data in the attribute you chose in the previous sub-step.

i **Note:** After an LDAP container is created, and resources have been created in that container, you cannot modify the "related primary/secondary attributes" that had been defined for the container.

j. Click .

k. Click **Save LDAP Source**.

A summary is displayed that shows the primary and secondary LDAP sources.

l. Optionally click **New LDAP Source** to add another secondary LDAP source, then repeat the sub-steps above to define the new secondary source.

m. Click **Next**.

14. Specify any desired resource attribute mapping, as described below.

You may need to *map* one or more resource attributes to attributes in the LDAP

source you have defined in your LDAP container. You may need to do this because the business process does not have direct access to the attributes in the LDAP source, but it does have access to the resource attributes in the organization model.

For the example shown in these steps, assume that the `mail` attribute in the LDAP source contains data that the business process needs. The business process has been designed in such a way to expect this data in the `EmailAddress` resource attribute. Therefore, if the `mail` LDAP attribute is mapped to the `EmailAddress` resource attribute, the business process will be able to access the user's email address at runtime.

- a. Select the version of the organization model that contains the attributes you want to map.
- b. From the **Select resource attributes** field, select an attribute from the organization model.
- c. From the **Select LDAP attributes** field, select the LDAP resource attribute you want to map to the attribute you selected in the previous step.

For example:

Map resource attributes

Select org model version

Select version

1 ▼

Select resource attributes ▼

Resource attribute name	LDAP resource attribute list
Email Address	mail ▼ ⊖

- d. Repeat the previous two steps to map additional attributes, if desired.
Note that to remove previously mapped attributes, click ⊖ to the right of the attributes you want to remove.
- e. When you've mapped all of the desired attributes, click **Next**.

15. Click **Create LDAP Container**.

- Click **Show Resources** to see a list of the resource candidates in the new LDAP container, or the x to the right of the **New LDAP Container** to see a list of all LDAP containers.

LDAP Query String Characters and Examples

Special characters can be used in LDAP queries when defining LDAP containers.

Special Character	Meaning
*	Wildcard character. Matches zero or more of any characters.
&	<p>Logical AND. Returns resources that satisfy the first string AND the second string.</p> <p>Place this special character to the left of the first query string, then enclose the entire expression in parentheses, as follows:</p> <pre>(&(string1)(string2))</pre>
	<p>Logical OR. Returns resources that satisfy the first string OR the second string.</p> <p>Place this special character to the left of the first query string, then enclose the entire expression in parentheses, as follows:</p> <pre>((string1)(string2))</pre>
!	<p>NOT. This means that you want all resources that do NOT match the specified value.</p> <p>Place this special character to the left of the query string to which it applies, inside of the parentheses:</p> <pre>(!(string))</pre>

Examples

The following are examples of LDAP query strings:

- The following query returns all resources that have sn attribute values beginning with “s”:

```
(sn=s*)
```

- The following query returns all resources that have sn attribute values beginning with “s” or “p”:

```
(|(sn=s*)(sn=p*))
```

- The following query returns all resources with carlicense attribute values equal to “Full” and employeetype attribute values equal to “Permanent”:

```
(&(carlicense=Full)(employeetype=Permanent))
```

- The following query returns all resources where sn attribute values *don’t* start with “s” and *don’t* start with “p”:

```
(&(!(sn=s*))(!(sn=p*)))
```

i Note: Depending on the specific LDAP server being used, the query syntax can vary. If the syntax described above does not return the expected results, consult the documentation for your LDAP server.

i Note: Values in a particular LDAP attribute may not be consistent across different LDAP servers.

Creating an LDAP Container using an LDAP Group

If you are using an LDAP group source in the LDAP container definition, a group DN is used to identify the directory entry that is the group. When a group DN is specified, a *member attribute* is also specified, which holds the collection of member identifiers (DNs). This provides the list of candidate resources.

Procedure

1. Open the Organization Browser (see [Accessing the Organization Browser](#)).

If any LDAP containers had previously been defined, they are shown in the left pane of the dialog box, and the details of the selected container are shown in the right pane.

If no LDAP containers have been created, the only thing displayed is the **Create New LDAP Container** button.

i Note: After you've defined a container, you can edit it by selecting the container in the list, then clicking  **>Edit**. You can generally follow the same steps in this procedure to edit an existing LDAP container.

However, if the existing LDAP container contains resources, the LDAP source for that container is fixed; you cannot delete or change the LDAP source.

Also note that if an LDAP source defined in an LDAP container is currently offline, you cannot edit the container until the LDAP source is back online.

2. Click **Create New LDAP Container**.
3. On the New LDAP Container dialog box, enter a name and an optional description for the new container.
4. In the **Select Organizations** section, you can optionally specify *organization relationships* for the new container (note that this is applicable only if there are multiple organizations in your organization model).

If the container has a relationship with an organization, resources in the container will be able to see that organization in the Organization Browser, as well as organizations that do not have an explicit relationship with a container. Resources can be mapped to positions in organizations that the user can see in the Organization Browser.

For more information about organization relationships, see [Container Organization Relationships](#).

If you are not specifying organization relationships for this container, proceed to the next step. (You can specify organization relationships for this container at a later time.)

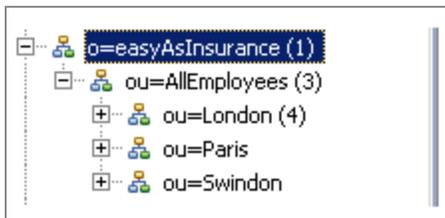
To specify organizational relationships, click the check box to the left of each organization for which the LDAP container is to have a relationship.

5. Click **Next**.
6. Ensure that the **Group source** option is selected.
7. In the **Alias** field, select the LDAP source from which you want to obtain resources.

The names in the **Alias** list are user-readable names that an administrator has assigned to each of the LDAP Servers available in the enterprise.

8. In the **Base dn** field, enter the branch (for example, an organization unit) in which you would like to limit the search in the LDAP directory structure. This increases the efficiency of the search if the LDAP contains a large number of branches.

The search base must provide the complete path to the desired branch in the LDAP directory structure. For example (this illustration is from an external application that shows the LDAP source), if you want to limit the search to the “London” organization unit in the following LDAP source ...



... you would enter the following in the **Base dn** field:

ou=London, ou=AllEmployees, o=easyAsInsurance

You can also leave the **Base dn** field blank, which causes the entire LDAP directory structure to be searched.

9. In the **Group DN Query** field, identify the objectClass of the LDAP directory whose entries are group entries.

This field defaults to "(objectClass=group)". Note that the value must be enclosed in parentheses.

For information about objectClasses, see [LDAP Group Sources](#).

10. In the **Resource name attribute(s)** field, enter one or more LDAP attributes by which you want the resources to be displayed in the list of candidate resources.

The resource name attribute is significant for the following reasons:

- It specifies the name by which the user must log in to the BPM application.
- It specifies the name by which the resource is listed when mapping resources to groups and/or positions in the organization model. That is, they must be names that the user doing the mapping can use to uniquely identify the resources. For example, you probably wouldn't want to use only "sn" (surname), as that may not be unique among all resources.

For example, the following shows resources when resource attributes were chosen to identify the resources by both forename and surname:

<input type="checkbox"/>	Select all resources
<input type="checkbox"/>	+  Clint
<input type="checkbox"/>	 John
<input type="checkbox"/>	+  Jon
<input type="checkbox"/>	+  Leon
<input type="checkbox"/>	 Liam Lawrence
<input type="checkbox"/>	 Richard
<input type="checkbox"/>	 Steve
<input type="checkbox"/>	 Tony

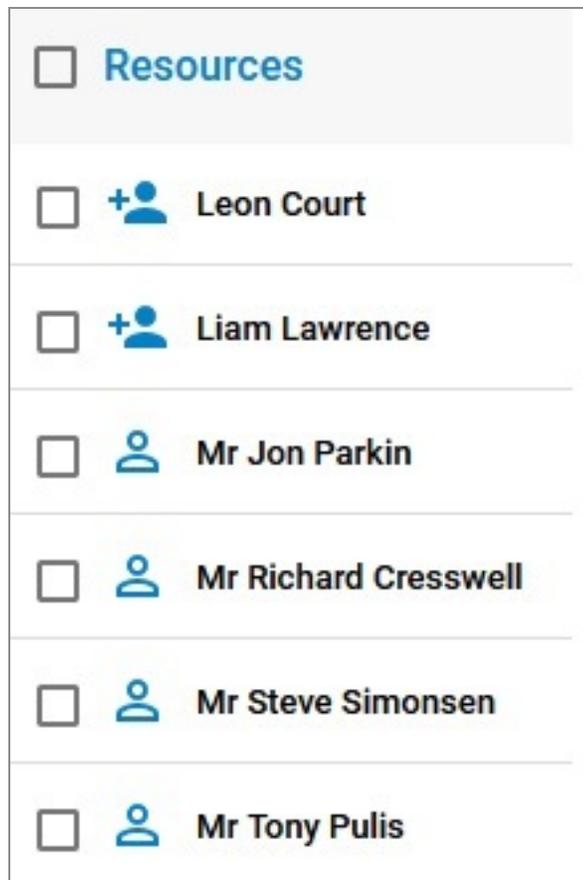
The default resource attribute is “cn”, which typically contains a full name. But depending on the data in the LDAP source, there may be more suitable attributes for this use.

You can specify multiple attributes in the **Resource name attribute(s)** field. For instance, you could enter “givenname sn” to display the resource’s first name and last name (again, depending on what is stored in those attributes on the chosen LDAP source).

Once you save the LDAP container you will be able to view the list of resources for the container. The LDAP entities that you see in this list, but have not been “added”, the resource name will be constructed based on the resource name attribute(s).

After a resource has been added (either using the **Add Resource** function or by mapping the user to a group or position), you can edit a resource's resource name by selecting the resource, then click .

It is possible to change the resource name attribute setting for the container, but that will not affect the resource name of resources that have already been added. It will, however, change the name that is constructed for the remaining LDAP entries that have not yet been added. For instance, using the example shown above, if you change the value in the **Resource name attribute(s)** field to “cn” (which contains "Mr" or "Mrs" with the resource's full name), the resource names now appear as shown below in the list of candidate resources.



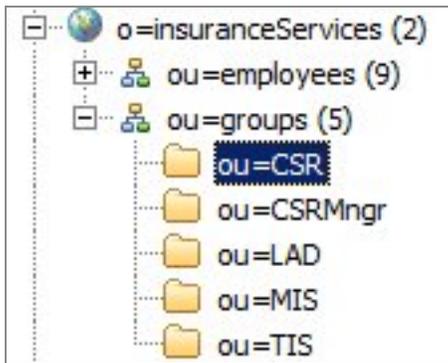
Notice that the resources that had been previously added (those that have the), are shown with the resource names they had when they were added; the resources that have not been added yet (blank check box), are shown with the new resource names.

11. Using the **Search Scope** options, specify the depth to perform the search in the LDAP directory structure, as follows:
 - **ONE LEVEL** - Only the elements directly within the Base-DN level are searched.
 - **SUBTREE** - Elements directly within, and below, the Base-DN level are searched.
12. Click **Fetch Group DN**.

This causes the Organization Browser to retrieve all of the group DNs whose objectClass matches the one specified in the **Group DN Query** field.

13. In the **Group dn** field, select the LDAP directory that contains the group entries.
For example, if the LDAP directory shown in the following illustration identifies the

group, the group DN is "ou=CSR,ou=groups,o=insuranceServices":



14. In the **Member attribute** field, select the attribute within the LDAP group entry that holds the collection of DNs that identifies the candidate resources.

For example, if the DN specified in the **Group dn** field contains the following attributes, where the roleoccupant attribute contains the DNs of the group members, you would specify "roleoccupant" as the member attribute:

DN: ou=CSR,ou=groups,o=insuranceServices	
Attribute Description	Value
objectClass	organizationalRole (structural)
objectClass	top (abstract)
cn	Customer Services Representative
ou	CSR
roleoccupant	uid=jparkin,ou=Employees,o=insuranceServices
roleoccupant	uid=rcrewell,ou=Employees,o=insuranceServices

15. Click **Fetch Sample** to view the group source sample, then click **Close** to close the sample dialog.

16. Click **Save LDAP Source**.

A summary of the LDAP source is displayed.

17. Optionally click **New LDAP Source**.

This optional step is used to define a secondary LDAP source. You can define one or more secondary LDAP sources. For more information, see [Primary and Secondary Sources](#).

- a. Ensure that **Group source** is selected as the source type.

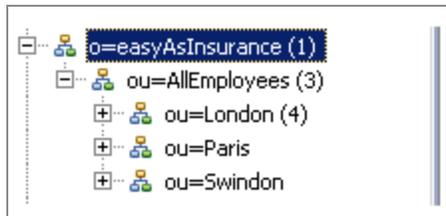
- b. From the **Alias** field dropdown list, select the secondary LDAP source you would like to add to the LDAP container.

Note that when choosing a secondary source, the alias that was chosen for the primary source is omitted from the **Alias** field dropdown list.

- c. In the **Base dn** field, enter the branch (e.g., an organization unit) in which you would like to limit the search in the LDAP directory structure.

This increases the efficiency of the search if the LDAP contains a large number of branches.

The search base must provide the complete path to the desired branch in the LDAP directory structure. For example, if you want to limit the search to the “London” organization unit in the following LDAP...



... you would enter the following in the **Base dn** field:

ou=London, ou=AllEmployees

- d. In the **Group DN Query** field, identify the objectClass of the LDAP directory whose entries are group entries.

This field defaults to "(objectClass=group)". Note that the value must be enclosed in parentheses.

For information about objectClasses, see [LDAP Group Sources](#).

Note: The **Resource name attribute(s)** field has no meaning when you are defining a secondary LDAP source.

- e. Using the **Search scope** options, specify the depth to perform the search in the LDAP directory structure, as follows:

- **ONE LEVEL** - Only the elements directly within the Base-DN level are

searched.

- **SUBTREE** - Elements directly within, and below, the Base-DN level are searched.

f. Click **Fetch Group DN**.

This causes the Organization Browser to retrieve all of the group DNs whose objectClass matches the one specified in the **Group DN Query** field.

- g. In the **Group dn** field, select the LDAP directory that contains the group entries.
- h. In the **Member attribute** field, select the attribute within the LDAP group entry that holds the collection of DNs that identifies the candidate resources.
- i. Click **Fetch Sample** to view the group source sample, then click **Close** to close the sample dialog box.
- j. At this point, you need to determine which attributes in the secondary LDAP source will be compared to attributes in the primary LDAP source.

The goal of comparing the primary and the secondary attributes is to ensure that data from the secondary LDAP source is only merged with the appropriate candidate resource from the primary LDAP source.

Where a match cannot be found, or where it is not one-to-one, the candidate resource will not have a complete, accurate set of information, and it will be either omitted (where no match is found) or marked as "multiple entries" (where the match isn't one-to-one).

For example, if there are several resources with the same last name, you need to check more than just the last name. In some cases, checking the first name will suffice. However, in other cases, you will need to check more attributes (because there may be multiple resources in the secondary LDAP source with the same first and last name — the system would not know which one to include). Maybe there are other types of data, such as an employee ID that would work better and would avoid inconsistencies in data entry (typos, nicknames, abbreviations, and more).

You may need to also go back and view the data in the attributes in the primary LDAP source.

For example, if you know that the "ou" attribute in the primary LDAP source

contains the complete name of the resources (Clint Hill, or John Eustace), and the “displayname” attribute in the secondary LDAP source contains the same information, those attributes would be prime candidates to link.

i **Note:** If you choose attributes that contain names, always be aware that there may be differences in the way those names were entered in the different LDAP sources, for example, Bob vs. Robert, Mike vs. Michael, or simple misspellings. Things like employee numbers tend to make good attributes to link.

For additional information, see [Primary and Secondary Sources](#).

- k. In the **primary attribute** field, choose the primary LDAP source attribute (for example, "ou") that contains data you want to compare to the data in the attribute you will choose in the next sub-step.
- l. In the **secondary attribute** field, choose the secondary LDAP source attribute (for example, "displayname") that contains data you want to compare to the data in the attribute you chose in the previous sub-step.

i **Note:** After an LDAP container is created, and resources have been created in that container, you cannot modify the "related primary/secondary attributes" that had been defined for the container.

- m. Click .
 - n. Click **Save LDAP Source**.
A summary is displayed that shows the primary and secondary LDAP sources.
 - o. Optionally click **New LDAP Source** to add another secondary LDAP source, then repeat the sub-steps above to define the new secondary source.
 - p. Click **Next**.
18. Specify any desired resource attribute mapping, as described below.

You may need to *map* one or more resource attributes to attributes in the LDAP source you have defined in your LDAP container. You may need to do this because the business process does not have direct access to the attributes in the LDAP

source, but it does have access to the resource attributes in the organization model.

For the example shown in these steps, assume that the `mail` attribute in the LDAP source contains data that the business process needs. The business process has been designed in such a way to expect this data in the `EmailAddress` resource attribute. Therefore, if the `mail` LDAP attribute is mapped to the `EmailAddress` resource attribute, the business process will be able to access the user's email address at runtime.

- a. Select the version of the organization model that contains the attributes you want to map.
- b. From the **Select resource attributes** field, select an attribute from the organization model.
- c. From the **Select LDAP attributes** field, select the LDAP resource attribute you want to map to the attribute you selected in the previous step.

For example:

Map resource attributes

Select org model version

Select version

1

Select resource attributes

Resource attribute name	LDAP resource attribute list
Email Address	mail ⊖

- d. Repeat the previous two steps to map additional attributes, if desired.
Note that you can remove previously mapped attributes by clicking ⊖ to the right of the attributes you want to remove.
 - e. When you've mapped all of the desired attributes, click **Next**.
19. Click **Create LDAP Container**.
 20. Click **Show Resources** to see a list of the resource candidates in the new LDAP container, **Edit** to edit the newly created container, or the x to the right of **New LDAP**

Container to see a list of all LDAP containers.

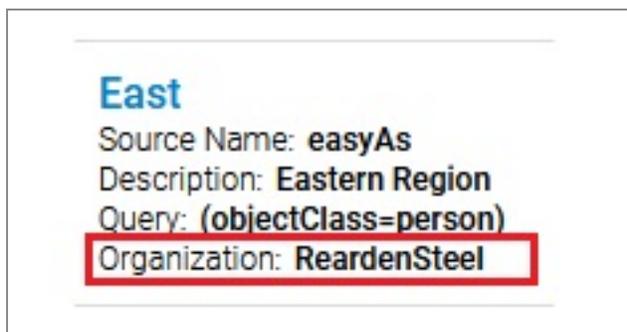
Container Organization Relationships

When you are creating or editing an LDAP container, you can specify that the LDAP container has a relationship with one or more organizations. These *organization relationships* allow you to prevent users from seeing LDAP containers and organizations they are not intended to see, as well as prevent resources from being mapped to positions in organizations they should not be in.

The ability to see containers and organizations (that is, the resources in those containers and organizations) has an impact when you are using the Organization Browser. It also affects the resources that you see when using the Reallocate Work Items to World function in a client application.

Note: Organization relationships do not apply to groups. That is, you cannot prevent a resource from seeing the resources in a particular group when using the Organization Browser or reallocating work items to the world.

If an organization relationship exists for the selected container, it is shown in the **Organizations** field on the Organization Browser's LDAP Containers dialog:



In this example, the LDAP container named "East" has a relationship with the ReardenSteel organization.

For more information about assigning organization relationships, see [Creating an LDAP Container using an LDAP Query](#) and [Creating an LDAP Container using an LDAP Group](#).

Overriding Organization Relationships

There is a system action called `Organization Admin` that can be used to override organization relationships in the following ways:

- Users who are assigned this system action can **see all containers, organizations, and resources**, regardless of the organizational relationships that are defined (you also need the `Browse Model` and `LDAP Admin` system actions to view LDAP containers).
- Users who possess this system action **can be mapped to any organization**, regardless of the organizational relationships that are defined.

Organization Relationship Examples

An organization relationship allows you to prevent users from seeing LDAP containers and organizations they are not intended to see, as well as prevent resources from being mapped to positions in organizations they should not be in.

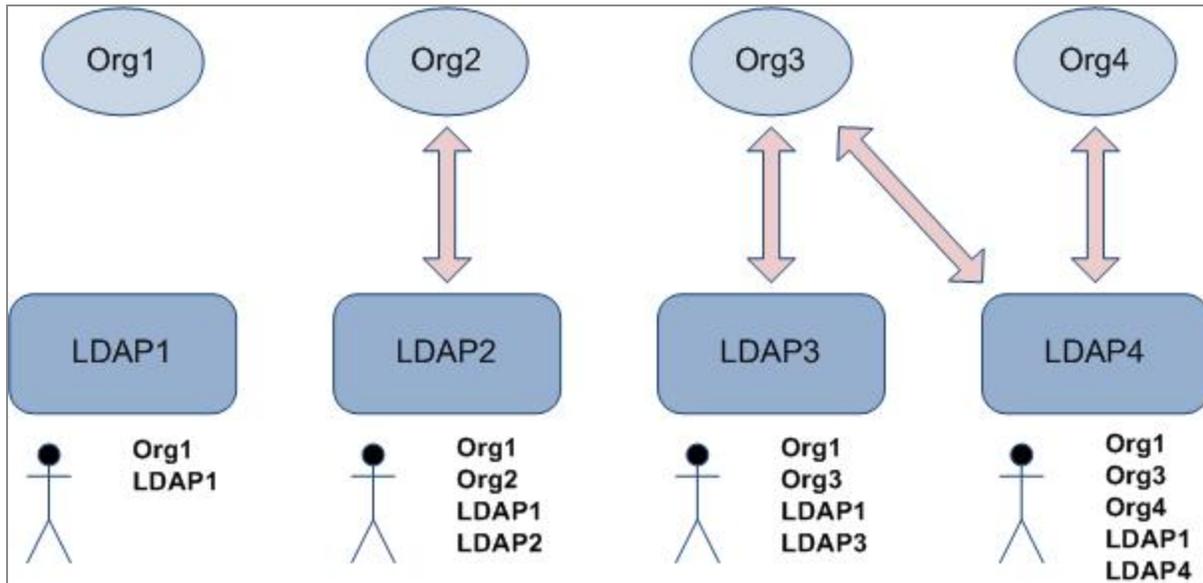
The descriptions that follow assume the resource does not have the `Organization Admin` system action.

The following summarizes the result of organizational relationships:

- Resources that are in a container that does not have a relationship with any organization can:
 - see containers that do not have a relationship with any organization.
 - see organizations that do not have a relationship with any container.
 - be mapped only to organizations that do not have a relationship with any container.
- Resources that are in a container that has one or more organization relationships can:
 - see containers that do not have a relationship with any organization, as well as the container they are in.
 - see organizations that do not have a relationship with any container, as well as organizations to which their container has a relationship.
 - be mapped to organizations that do not have a relationship with any container, as well as to the organizations that have a relationship with the container the

resource is in.

The following graphic illustrates these points by showing four organizations and four LDAP containers. The arrows represent a relationship between the container and the organization. Under each container is a resource that is in that container, and to the right of each resource is shown the organizations and LDAP containers the resource can see.



- All resources can see Org1 and LDAP1 because neither has an explicit relationship set up.
- The resources in containers that have an organizational relationship can also see the LDAP container they are in, as well as the organizations for which their container has a relationship.
- Any of the resources can be mapped to Org1, as well as to the organization(s) to which their container has a relationship.

An important point to understand here is that if multiple containers have a relationship with a single organization, the resources in one container will *not* be able to see the resources from the other container when viewing the organization to which they both have a relationship.

For example, using the illustration above, if a resource from both LDAP3 and LDAP4 are mapped to the same position in Org3, when a resource from LDAP3 looks at that position (using the Organization Browser, or when reallocating work items to the world), that resource will *not* see the LDAP4 resource that is mapped to that position. Likewise, a resource from LDAP4 will not see the LDAP3 resource when looking at that position.

Invalid Mappings Because of Organization Relationships

If you are logged in as a user who has the `Organization Admin` system action, you can see all organizations regardless of the organization relationships that are defined. However, the system does not allow you to map resources to positions in organizations to which the resource is barred because of organization relationships.

If a resource is mapped to a position, then an organization relationship is defined that bars the resource from the organization, the resource remains mapped to the position; the resource is not automatically un-mapped from the position. You can manually remove the mapping using the `Remove` function, if desired.

Deleting LDAP Containers

LDAP containers can be deleted using the `Organization Browser`.

To delete LDAP containers, you must be assigned the `Delete Resource` Admin system action and the `Delete LDAP` Admin system action.

If the container contains BPM resources, those resources are automatically removed from any groups or positions to which they were mapped.

Procedure

1. Display the list of LDAP containers.
2. Hover the mouse pointer over the LDAP container that you want to delete, then click .
3. Click **Delete**.
4. On the confirmation dialog, click **Delete**.

Resources

Resources represent users who log into BPM applications and receive work items that they need to work on.

Resources are obtained from *LDAP sources*, which are separate servers to which the `Organization Browser` connects to get a list of *candidate resources*. Before a resource can

log into a BPM application, the candidate resource must be made a *BPM resource* by using the Add Resource function in the Organization Browser.

Resources can be mapped to groups or positions in the organization model; resources receive work items in their work list because of being mapped to certain groups or positions. If you map a candidate resource to a group or position, the Add Resource function is run before mapping, making that resource a BPM resource, before it is mapped.

If you just "add" a resource, without mapping the resource to a group or position, it allows the resource to log in to BPM applications, but the resource will not receive work items until after being mapped to a group or position in the organization. Resources can also be deleted, preventing them from logging in.

Resources can log into the BPM application using the "Resource Name" assigned to the resource (which can be modified), and a password specified in the LDAP source.

i Note: By default, BPM applications recognize the username "tibco-admin" with a password of "secret" as the System Administrator (although these names may be changed). This is the only user authorized to log in until another user is configured using the Organization Browser. By default, the tibco-admin resource is in the System Admin container.

Adding BPM Resources

A candidate resource must be "added" to become a BPM resource. This results in an entry for that resource being added to the BPM database. Once a resource is a BPM resource, that resource can log into BPM applications.

BPM resources can be created in the following ways:

- Map the resource to a group or position. This automatically causes the resource to become a BPM resource.
See [Mapping Resources](#).
- Use the Add Resource function. This function allows you to add the resource to the database without mapping the resource to a group or position, allowing the user to log into BPM applications.

There are a couple of ways in which you can add resources. One can be used to add one or more resources at one time. The other can be used to add a single resource. If you are adding a single resource using the second procedure shown below, you can

also modify the resource's Resource Name (which is what the resource uses to log in), if desired.

Adding one or more resources:

1. Select the LDAP container in which the resources are candidate resources.
2. In the list of resources, select the resources you want to add by checking the box to the left of the resources.
3. Click **Add Selected**.
4. On the **Add resource(s)** dialog, modify the **Resource name** and/or **Resource label** for one or more of the resources, if needed.

For more information, see [Renaming a Resource](#).

5. Click **Create resource**.

Adding a single resource:

1. Select the LDAP container in which the resource is a candidate resource.
2. In the list of resources, hover the mouse pointer over the resource you want to add and click **Add resource**.
3. On the **Add resource(s)** dialog, modify the **Resource name** and/or **Resource label**, if needed.

For more information, see [Renaming a Resource](#).

4. Click **Create Resource**.

BPM resources can also be deleted. When a BPM resource is deleted, it once again becomes a candidate resource; that resource can no longer log into BPM applications. For more information, see [Deleting Resources](#).



Note: You must have the Create Resource Admin system action to add or edit resources.

Mapping Resources

Mapping resources is an administrative task that will typically be performed before users start using BPM applications.

Mapping resources can then be performed on an ongoing basis as resources are added or removed from the system, or if the organization model is revised (new positions, groups,

and so on).

Mapping resources involves assigning resources to specific groups and/or positions in an organization model, which results in the resources receiving work items that are sent to the groups/positions to which the resources have been mapped.

Resources can be mapped to groups and/or positions:

- A *group* represents a job type within the organization. It allows resources to be grouped by their job characteristics.

Groups are hierarchical, that is, you can have parent groups with subgroups. Typically, sub-groups are specializations of the parent group. For example:



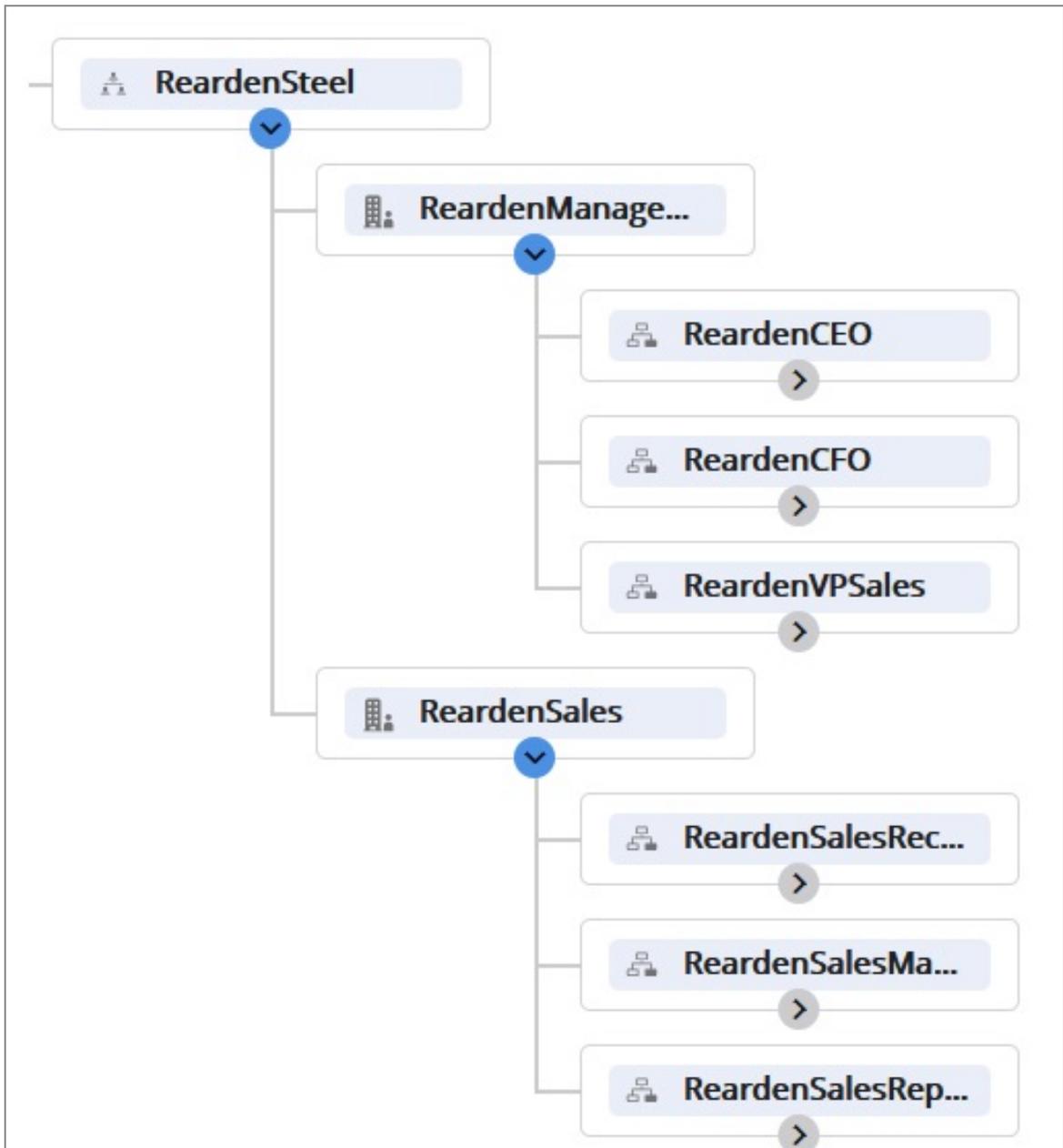
- If a group has sub-groups, you can assign resources to either the parent group or the sub-group. Resources that belong to sub-groups receive work items that are offered to their parent groups, as well as to the sub-group to which they belong. Using the example above, resources in the WorkItemManagement and ResourceManagement groups will also receive work items offered to the GlobalWorkManagement group.

For information about the inheritance of privileges and capabilities assigned to groups, see [Privileges](#) and [Capabilities](#).

- A *position* represents a set of responsibilities for a job within an organization unit. It allows resources to be grouped by job responsibility.

Positions are subordinate to an organization unit in the organization model. An organization unit can have many positions. Positions cannot be nested, although organization units can. In the following example, ReardenManagement and

ReardenSales are organization units under the ReardenSteel organization. Each of the organizational unit has three positions.



Resources can be mapped to positions, but not organizations or organization units.

Resources that belong to a position receive work items offered to the position, as well as work items offered to the organization unit that is the immediate parent of the position.

For information about the inheritance of privileges and capabilities assigned to positions, see [Privileges](#) and [Capabilities](#).

Procedure

1. Select the LDAP container that contains the candidate resources you want to map.
2. In the right pane, click either **ORGANIZATIONS** or **GROUPS**, depending on whether you want to map resources to a position or group.
3. Expand the organization or group structure to expose the position or group to which you want to map a resource.
4. Map resources to the position or map in one of the following ways:
 - Drag a resource into the desired position or group.
 - Select one or more resources by clicking the box to the left of the resource, then click **Map Selected Resources** in the position or group.
 - Select one or more resources by clicking the box to the left of the resource, then choose any of the selections from the Actions menu in the upper-right part of the screen:
 - Map to all positions and groups
 - Map to all positions
 - Remove from all positions and groups
 - Remove from all positions

If one or more of the resources had not been added to the system or has not been mapped to a position or group, the Add and map resources dialog box is displayed.

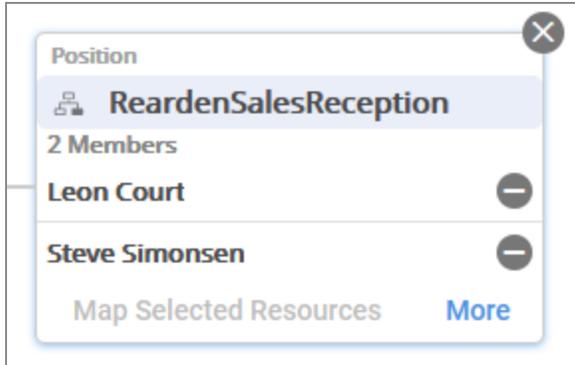
5. If the Add and map resources dialog box is displayed, optionally change the resource's resource name or label, then click **Add and map resources**.

If the group or position to which you are mapping the resource has capabilities defined, and the resource does not have those capabilities, a message is displayed indicating that. This is only informational; capabilities are not enforced.

For information about resource names and labels, see [Renaming a Resource](#).

Result

If you had dragged the resource into a position or group, the resource is now shown as a "member" of the position or group:



Removing Resources from Groups or Positions

Removing resources from groups or positions prevents the resources from receiving work items destined to the groups or positions.

Procedure

1. Navigate to the group or position in which the resource is a member.
2. Select the group or position.
3. Click  to the right of the resource you want to remove.

Viewing Resource Details

Resource details include things like groups and positions to which a resource is mapped, privileges held by the resource, and so on.

Procedure

1. Select the LDAP container that contains the resource whose details you want to view.
2. Select the desired resource.

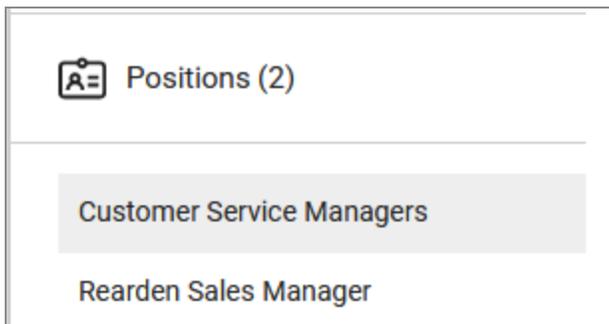
Select only BPM resources (resources that have  next to their name). No details are available for candidate resources.

The right pane lists the following details for the selected BPM resource:

- **Capabilities** - These are the capabilities possessed by the selected resource. For more information, see [Capabilities](#).

- **Groups** - These are the groups to which the resource has been mapped. For more information, see [Mapping Resources](#).
 - **Positions** - These are the positions to which the resource has been mapped. For more information, see [Mapping Resources](#).
 - **Privileges** - These are the privileges that the resource has inherited. The name in parentheses is the name of the group or position the user was mapped to that caused the resource to inherit the privilege. For more information, see [Privileges](#).
 - **Attributes** - This is the list of available resource attributes, as well as the value of each one for the selected resource. For more information, see [Viewing and Editing Resource Attributes](#).
 - **Push Destinations** - These are the push destinations that have been assigned to the selected resource. Push destinations specify the destination(s) to which work items sent to a resource are to be pushed. For more information, see [Configuring Resource Push Destinations](#).
 - **Location** - The location to which the resource is assigned. Locations are defined in the organization model.
3. Click any of the items to expand it and provide details.

For example:



Renaming a Resource

When users log into a BPM application, they must enter their *resource name*. Initially, each user's resource name is established as the value that is stored in an attribute in an LDAP source. The LDAP attribute is specified when you create an LDAP container.

After a resource is added to an LDAP container, you can specify a name different than the one stored in the LDAP source attribute.

Procedure

1. Select the LDAP container that contains the resource you want to rename.
2. Hover the mouse pointer over the resource whose resource name you want to change and click .
3. Click **Edit resource**.
4. From the **Update resource(s)** dialog box, you can modify the following:
 - **Resource name:** This is the name the resource must use to log into BPM applications.
 - **Resource label:** This is a label that is shown for the resource in various areas of the Organization Browser, such as the list of resources mapped to a particular position or group.
 - **Location:** This location to which the resource is assigned. Locations are defined in the organization model.
5. Click **Update resource**.

Deleting Resources

BPM resources can be deleted. After deleting a resource, it becomes a candidate resource again. That resource can no longer log into the BPM application and is removed from any groups and positions the resource is a member.

Note that deleting a resource using the Organization Browser does not remove the resource from the LDAP source, or from the container. It becomes a candidate resource and deletes any mapping that may have been done for that resource, preventing that user from being able to log into BPM applications.

You must have the Delete Resource Admin system action to delete resources.

Procedure

1. Select the LDAP container containing the resource that you would like to delete.
2. Hover the mouse pointer over the desired resource, then click .
3. Click **Delete resource**.

Result

i Note: If you delete an LDAP container that contains BPM resources, the resources are deleted at the same time. For more information, see [Deleting LDAP Containers](#).

Managing Deleted Users

If a user is deleted from TIBCO BPM Enterprise, and the deleted user is the principal of an outstanding process instance, that process instance will eventually fail. This will occur because a process instance performs actions on behalf of the principal of the process instance. After a user is deleted, that user cannot perform any actions in TIBCO BPM Enterprise.

Therefore, if a user has been deleted, and that user has outstanding process instances, you must reassign those process instances to another user.

Using the Organization Browser, you can:

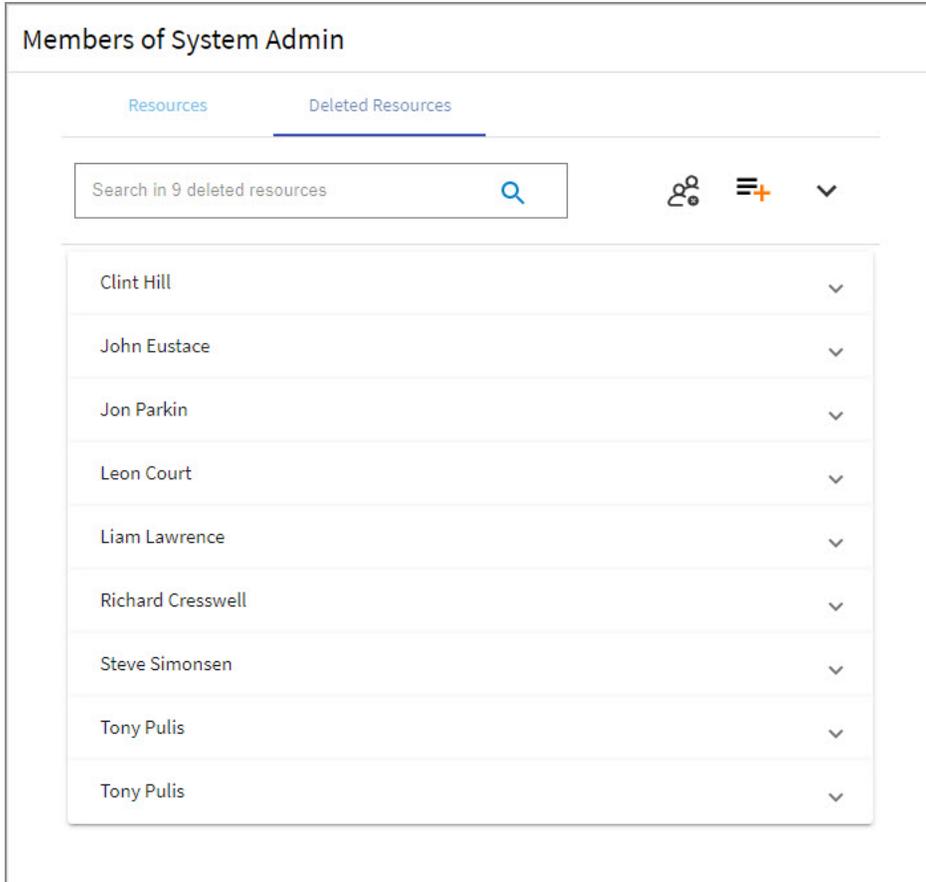
- View a list of deleted users - see [Viewing Deleted Users](#).
This requires the Resource Admin system action.
- View details about a deleted user - see [Viewing Details of Deleted Users](#).
This requires the Resource Admin system action.
- Purge deleted users - see [Purging Deleted Users](#).
This requires the Delete Resource Admin system action.

Viewing Deleted Users

Using the Organization Browser, you can view a list of resources that have been deleted from TIBCO BPM Enterprise.

Procedure

1. From the Organization Browser, select the View admin details.
2. Select the **Deleted Resources** tab:



What to do next

From the list of deleted users, you can:

- View details about a single deleted resource - see [Viewing Details of Deleted Users](#).
- Purge deleted users from the list - see [Purging Deleted Users](#).

Viewing Details of Deleted Users

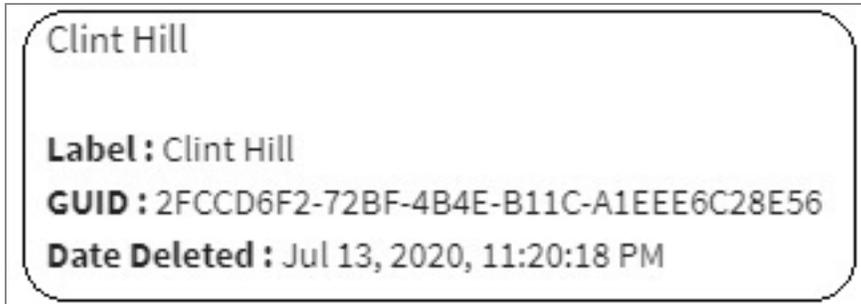
You can view information about deleted users, such as their GUID, the date they were deleted, and so on.

Note: Deleted users can be *purged* from the system as well; once purged, their details are no longer available using this procedure - see [Purging Deleted Users](#).

Procedure

1. Display the list of deleted users as described in [Viewing Deleted Users](#).
2. From the list of deleted users, select a single user.

Details of when the user was deleted is displayed. For example:



Purging Deleted Users

After users are deleted from TIBCO BPM Enterprise, you can also *purge* them from the deleted users list. This provides a method of managing the size of the Deleted Users list so that it doesn't become too large over time.

After being purged, the user is still a *candidate resource*, and can be created again (assuming the user is still in the LDAP source).

To purge deleted users, you must have system action, Delete Resource Admin.

Procedure

1. Display the list of deleted users as described in [Viewing Deleted Users](#).
2. Do one of the following:
 - Hover the mouse over the user you want to delete. When the  icon appears, click it.
 - Click **Purge All** to purge all users in the list.

Result

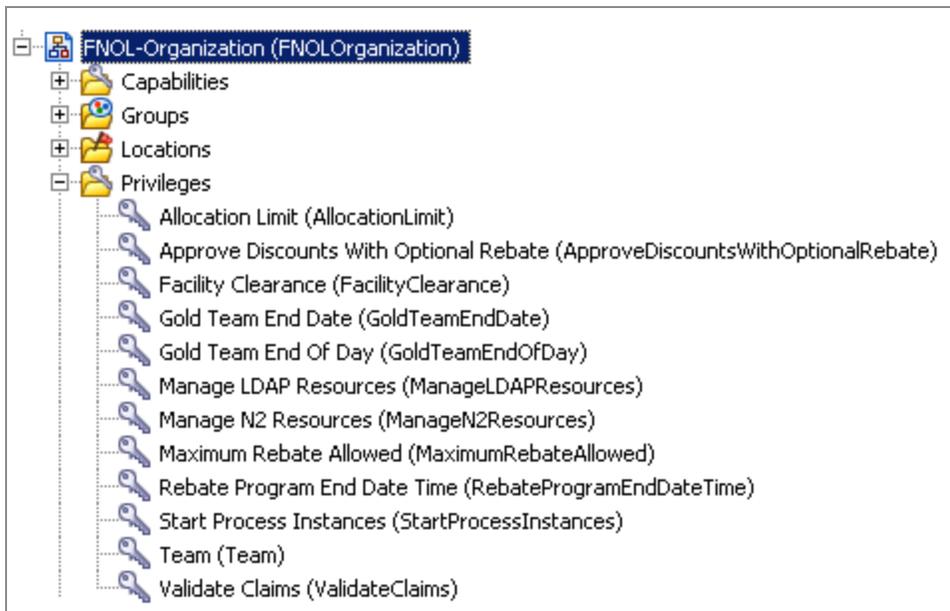
After being purged, the user is immediately removed from the list of deleted users.

Privileges

Privileges represent authorities. They are used by the BPM application to determine which functions the user can access in the application. Users obtain privileges by being mapped to groups and/or positions to which privileges have been assigned.

BPM applications use privileges in conjunction with *system actions* to determine access to functions in the application.

Privileges are defined in the organization model using the TIBCO Business Studio - BPM Edition Organization Modeler. The following is an example from TIBCO Business Studio - BPM Edition of some privileges that have been created in an organization model:



For more information, see "Creating Privileges" in the *TIBCO Business Studio™ - BPM Edition Application Designer's Guide*.

Privileges that have been defined in the organization model can be assigned to the following:

- **Groups, organization units, and positions** - Privileges can be assigned to these entities when they are created in the TIBCO Business Studio - BPM Edition Organization Modeler.

Resources that are mapped to these entities inherit the privileges of those entities, as follows:

- **Groups** - Members of groups inherit the privileges of the group, as well as the

privileges of *all* parent groups.

- **Positions/organization units** - Members of a position inherit the privileges of the position, as well as the organization unit that is the immediate parent of the position. If organization units are nested, members of the position do *not* inherit privileges from organization units further up the tree — only the immediate parent.

For information about viewing the privileges assigned to groups, organization units, and positions, see [Browsing Groups](#) and [Browsing Organization Units and Positions](#).

- **Resources** - Resources cannot be given privileges directly. They obtain them only by virtue of being a member of a group or position.

For information about viewing the privileges a resource has inherited by being mapped to groups or positions, see [Viewing a Resource's Privileges](#).

Viewing a Resource's Privileges

A resource is granted privileges by being mapped to groups or positions that have been assigned the privileges in TIBCO Business Studio - BPM Edition .

Procedure

1. Select the LDAP container containing the resource whose privileges you want to view.
2. Select the resource.

In the right pane, the number in parentheses to the right of "Privileges" indicates the number of privileges held by the resource.

3. In the right pane, click **Privileges** to view the privileges held by the resource.

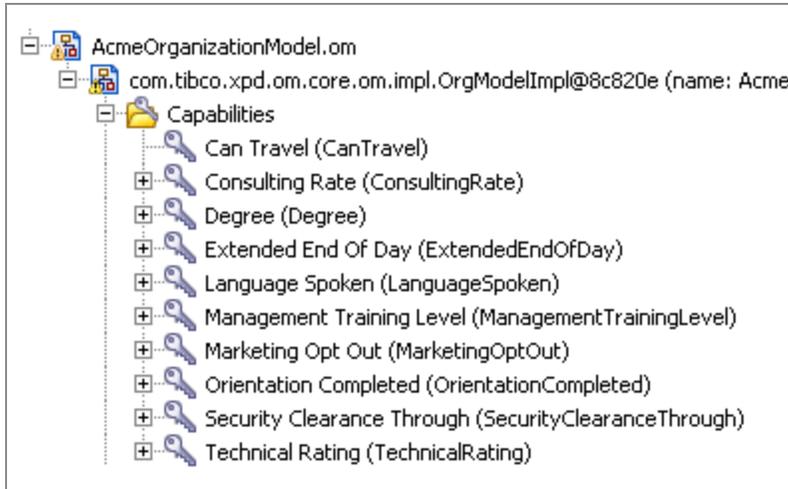
For example:



Capabilities

Capabilities represent skills needed to perform a task, for instance, being bilingual. They can also be further qualified, for instance, by specifying a specific language needed.

Capabilities are defined in the organization model using the TIBCO Business Studio - BPM Edition Organization Modeler. The following is an example of some capabilities that have been created in an organization model:



After capabilities are defined in the organization model, they can be assigned to the following:

- **Groups and positions** - Capabilities are assigned to groups and positions using the Organization Modeler. The purpose of this is to state that resources assigned to that group or position *should* have that capability.

For example, assume you have a position to which the “LanguageSpoken” capability with a qualifier of “German” has been assigned. When someone is mapping resources using the Organization Browser, they should map only resources that have the “LanguageSpoken/German” capability for that position.

Note, however, that this is *not* an enforced requirement — the Organization Browser will allow you to assign a resource who does not have the required capability to a group or position that has been assigned that capability.

For information about the capabilities assigned to groups and positions (this is, the capabilities that resources should be mapped to those groups and positions), see [Browsing Groups](#) and [Browsing Organization Units and Positions](#).

- **Resources** - Capabilities are assigned to individual resources using the Organization Browser. The purpose of this is to state that the resource has the capability, for

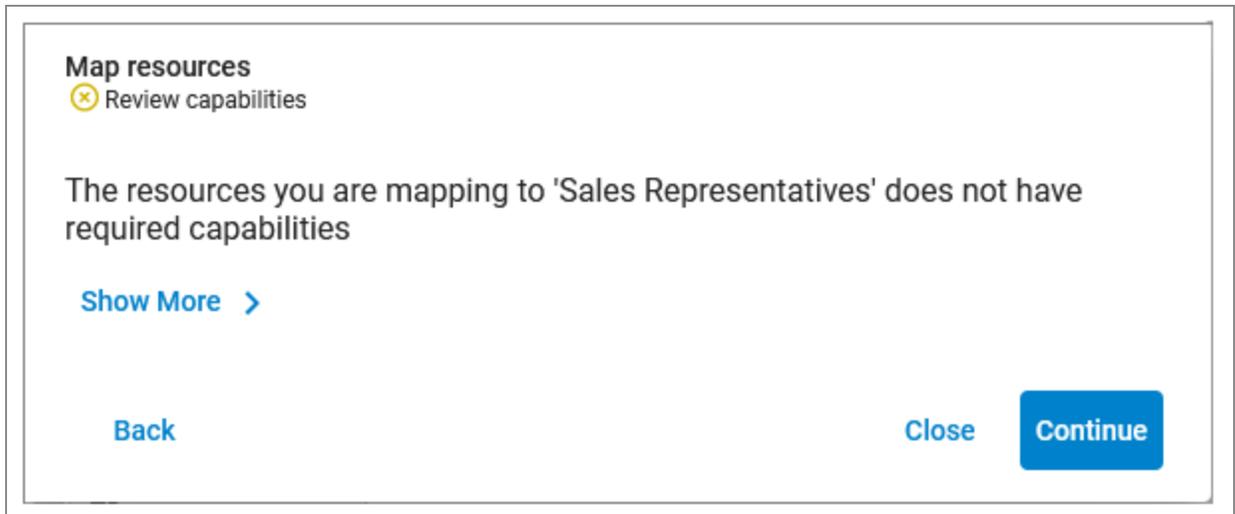
example, to speak a specific language.

As stated above, when resources are assigned to groups and positions that require capabilities, the assigner should ensure that only resources that have the required capabilities be assigned to those groups and positions.

Note that resources do *not* inherit capabilities based on their membership in groups and positions; capabilities must be assigned directly to resources using the Organization Browser.

For more information, see [Viewing and Editing Resource Capabilities](#).

If you attempt to map a resource to a group or position that has capabilities assigned to it, and the resource does not have the capabilities, a warning message is displayed in the Organization Browser. For example:



If you click **Show More**, it shows the capability needed by the group or position. For example:

Map resources
 Review capabilities

The resources you are mapping to 'Sales Representatives' does not have required capabilities

[Show Less](#) 

<p>Resources</p> <div style="background-color: #f0f0f0; padding: 5px; margin-bottom: 5px;">Tony Pulis</div>	<p>Required capabilities by 'Sales Representatives'</p> <div style="background-color: #f0f0f0; padding: 5px; margin-bottom: 5px;">Can Travel</div>
--------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------

Note, however, required capabilities are informational only; they are not enforced.

Viewing and Editing Resource Capabilities

Viewing and editing a resource's capabilities is done using the Organization Browser.

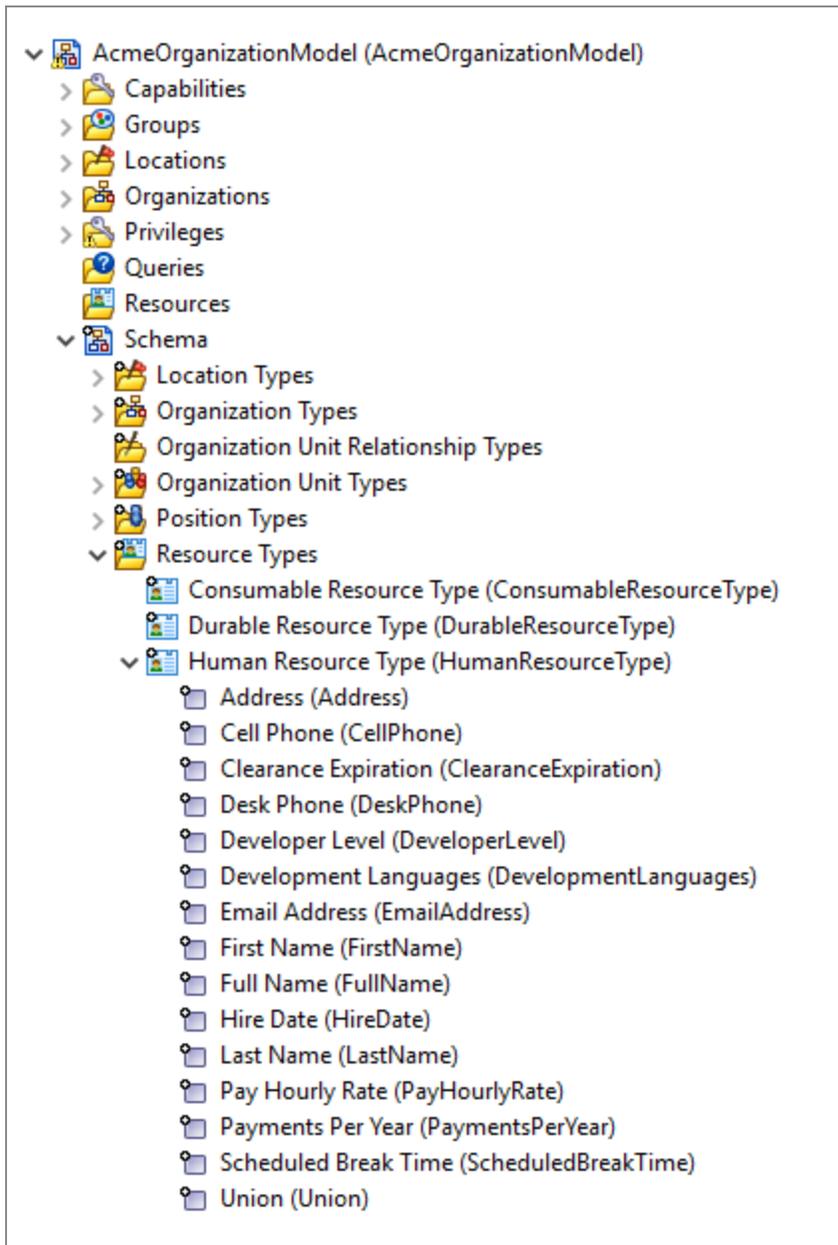
Procedure

1. Select the LDAP container containing the resource whose capabilities you want to view or edit.
2. Select the desired resource.
 Select only a BPM resource (a resource that has a  icon next to the resource name); candidate resources cannot have capabilities assigned to them.
3. In the right pane, click **Capabilities**.
 If any capabilities are assigned to the resource, they are displayed. Note that the number shown to the right of the header indicates the number of capabilities currently assigned to the resource.
4. To assign a capability to the resource, remove an assigned capability, or to edit a capability already assigned to the resource, click **Edit**.
 All of the capabilities that are defined in the organization model are listed.
5. Assign or remove the desired capabilities, then click **Save**.
6. Click the **Capabilities** header again to collapse the list of capabilities.

Resource Attributes

When an organization model is created using the TIBCO Business Studio - BPM Edition Organization Modeler, *resource attributes* can also be created. Resource attributes are used to store information about resources. For example, there may be an “EmailAddress” attribute defined, in which each resource’s email address is stored. These attributes can contain data that the business process may access during runtime.

The following shows an example of some resource attributes that were created in the Organization Modeler:



These attributes are available for each BPM resource. For example, each resource's cell phone number can be stored in the CellPhone attribute. You can use the Organization Browser to assign values to a resource's resource attributes - see [Viewing and Editing Resource Attributes](#).

You may also need to *map* one or more of these resource attributes to attributes in the LDAP sources you have defined in your LDAP container. You may need to do this because the business process does not have direct access to the attributes in the LDAP sources, but it does have access to the resource attributes in the organization model. When you map a resource attribute to an LDAP attribute, it gives the business process access to the data in the LDAP attribute at runtime. Resource attributes are mapped to LDAP attributes in an LDAP container definition. For information, see [Creating an LDAP Container](#).

For information about creating resource types, and adding attributes to the type, see "Creating a Resource Type" in the *TIBCO Business Studio™ - BPM Edition Application Designer's Guide*.

Viewing and Editing Resource Attributes

Values in the resource attributes for a resource can be modified using the Organization Browser.

Procedure

1. Select the LDAP container containing the resource whose privileges you want to view.
2. Select the resource.

In the right pane, the number in parentheses to the right of "Attributes" indicates the number of attributes that have been assigned to the Human Resource Type in the organization model.

3. In the right pane, click **Attributes** to view the available resource attributes.

For example:

Attributes (15)		Edit
Cell Phone	555-222-7785	
Clearance Expiration	2020-05-31T07:05:00-07:00	
Desk Phone		
Developer Level		
Development Languages		
Email Address	susieq@megacorp.com	
First Name	Susie	
Last Name	Que	
Full Name		

4. Click **Edit**.
5. Add a value, or edit an existing value, as needed.
6. Click **Save**.

Push Destinations

“Pushed” distribution of work items is supported. In a pushed distribution model, when a work item is generated, it is sent to a user as an email. The email contains the URL of the work item, which the user can click to open and process the work item.

Generally, work items are “pulled” when a user logs into a BPM application and accesses their work list — this “pulls”, from the TIBCO server, the work items that are assigned to that user.

Users do not need to log into the BPM application to access a pushed work item, although they are required to authenticate themselves before they can open the work item.

Note: For sending push destination emails, a configured SMTP shared resource is required. For information, see [SMTP Connection Shared Resources](#).

**Note:**

By default, the work item URL received in your email inbox (configured with the help of SMTP connection) points to localhost.

To receive a proper URL in your email inbox to open a work item, you must add the `formRedirectURL` property and its value in Work Presentation (via Configuration Management). For more details, see [Work Presentation Configuration](#).

A pushed distribution is useful for occasional users. For example, managers who only need to become involved in a process when some form of higher level approval is required. These users are typically not logged into the BPM application all the time, and so could otherwise miss the arrival of high-priority work items.



Note: If you use the presentation channel settings (push destinations) to deliver notification of work items via email, on the **Work Resource** tab for the user task, you must set the Distribution Strategy to **Allocate to One** rather than **Offer to All**. For example, if you have a performer field set to: `resource(name='susieq')`, `susieq` receives an email notification of a work item only if the Distribution Strategy is **Allocate to One**.

The `Write Push Destinations` system action is required to configure push destinations.

Push destinations can be specified for:

- **Organizational entities** - These specify the destination(s) to which work items sent to the organizational entity are to be pushed. You can specify one or more push destinations for each organizational entity. For information, see [Configuring Organizational Entity Push Destinations](#).
- **Resources** - These specify the destination(s) to which work items that are sent *directly* to the resource are to be pushed. You can specify one or more push destinations for each resource. For information, see [Configuring Resource Push Destinations](#).

Configuring Organizational Entity Push Destinations

Work items can be *pushed* to an organizational entity, rather than be *pulled* from a work item list. The organizational entity push destination defines how, and to where, the work items are pushed.

Procedure

1. From the Organization Browser, click **Browse Organization**.
2. Click either:
 - **ORGANIZATIONS** to edit a push destination for an organization unit or position.
 - **GROUPS** to edit a push destination for a group.
3. From either the groups or organizations structure, select the desired group, organization unit, or position, then click **More**.
4. In the right pane, click **Push Destinations**.

If any push destinations are assigned to the organizational entity, they are displayed. Note that the number shown to the right of the header indicates the number of push destinations currently assigned to the organizational entity.

5. Configure a push destination as follows:
 - a. To add a new push destination, click **Add Destination**, then complete the Add Push Destinations dialog box according to the field descriptions below.
 - b. To edit an existing push destination, hover the mouse pointer over the desired destination, click , then complete the Edit Push Destination dialog box according to the field descriptions below.
 - c. To delete an existing push destination, hover the mouse pointer over the desired destination, then click .

Field	Description
Name	A descriptive name for the push destination.
Channel Type	The type of channel that will be used to push work items to the organizational entity. Currently, the only available channel type is “emailchannel”, which causes work items to be pushed to an email address.
Channel Id	Uniquely identifies the presentation channel to use when pushing work items to the organizational entity. The channel ID is defined when a presentation channel is defined in TIBCO Business Studio - BPM Edition.

Field	Description
	For more information, see "Configuring Presentation Channels" in the <i>TIBCO Business Studio™ - BPM Edition Application Designer's Guide</i> .
Target	The email address to which work items are to be pushed.
Active	Check the box to make the push destination active. This provides a means of disabling the push destination without removing it.

6. Click **Save** to save the push destination definition.

Configuring Resource Push Destinations

Work items can be *pushed* to a resource, rather than be *pulled* from a work item list. The resource push destination defines how, and to where, the work items are pushed.

Procedure

1. Select the LDAP container containing the resource whose privileges you want to view.
2. Select the desired resource.

In the right pane, the number in parentheses to the right of "Push Destinations" indicates the number of push destinations that have been assigned to the resource.

3. In the right pane, click **Push Destinations**.

If any push destinations are assigned to the resource, they are displayed. In the right pane, the number in parentheses to the right of the header indicates the number of push destinations that have been assigned to the resource.

4. Configure a push destination as follows:
 - a. To add a new push destination, click **Add Destination**, then complete the Add Push Destinations dialog box according to the field descriptions below.
 - b. To edit an existing push destination, hover the mouse pointer over the desired destination, click , then complete the Edit Push Destination dialog box according to the field descriptions below.
 - c. To delete an existing push destination, hover the mouse pointer over the desired destination, then click .

Field	Description
Name	A descriptive name for the push destination.
Channel Type	The type of channel that will be used to push work items to the resource. Currently, the only available channel type is “emailchannel”, which causes work items to be pushed to an email address.
Channel Id	Uniquely identifies the presentation channel to use when pushing work items to the resource. The channel ID is defined when a presentation channel is defined in TIBCO Business Studio - BPM Edition. For more information, see "Configuring Presentation Channels" in the <i>TIBCO Business Studio™ - BPM Edition Application Designer's Guide</i> .
Target Source	Using the dropdown list in this field, you can select a resource attribute that contains the email address. You can either use this field or the Targets field (see below) to specify the email address. If you specify a source for the email address in the Target Source field, the Targets field is disabled. Note that if the email address is ultimately coming from an LDAP attribute, you must map the LDAP attribute to a resource attribute when the LDAP container is created — for information, see Creating an LDAP Container .
Target	The email address to which work items are to be pushed. This field is not displayed if an attribute is specified in the Target Source field.
Active	Check the box to make the push destination active. This provides a means of disabling the push destination without removing it.

5. Click **Save** to save the push destination definition.

Dynamic Organization Model Extension Points

An organization model extension point is a dynamic organization unit that references a dynamic organization template.

The extension point configuration is used to dynamically generate instances of the organization model template directly below it.

The designation of the extension point is a design-time function; it consists only of the assignment of the organization model template. The remaining extension point configuration — that is, the LDAP connection information — is not known, nor can it be interrogated, at design-time, and is therefore performed after deployment using the Organization Browser (it can also be done using the API).

Note that any extension point that is not fully configured is ignored, and does not result in the creation of organization model template instances.

After an extension point is configured, instances of the organization model template are generated using the following Directory Engine properties:

- `extensionPointProcessEnable` - Enables (true) or disables (false) the generation of instances of dynamic organization models each day at the time specified in the `extensionPointProcessStart` property.
- `extensionPointProcessStart` - Specifies the time each day to generate instances of dynamic organization models as long as `extensionPointProcessEnable` is set to true.
- `extensionPointProcessInterval` - The delay between the start of one extension point processing event and the next. This value should be great enough to ensure that two events do not overlap. The value is expressed as an XML Schema Duration string.
- `extensionPointDeleteEnabled` - Enables (true) or disables (false) the removal of previously generated instances of dynamic organization models if the LDAP entry/attribute from which they were derived has been removed from the LDAP source.

For more information about Directory Engine properties, see [Directory Engine Configuration](#).

For more information about dynamic organizations, see "Dynamic Organizations" in the *TIBCO Business Studio - BPM Edition Application Designer's Guide*.

Configuring Dynamic Organization Model Extension Points

An extension point configuration is used to dynamically generate instances of the organization model template directly below it.

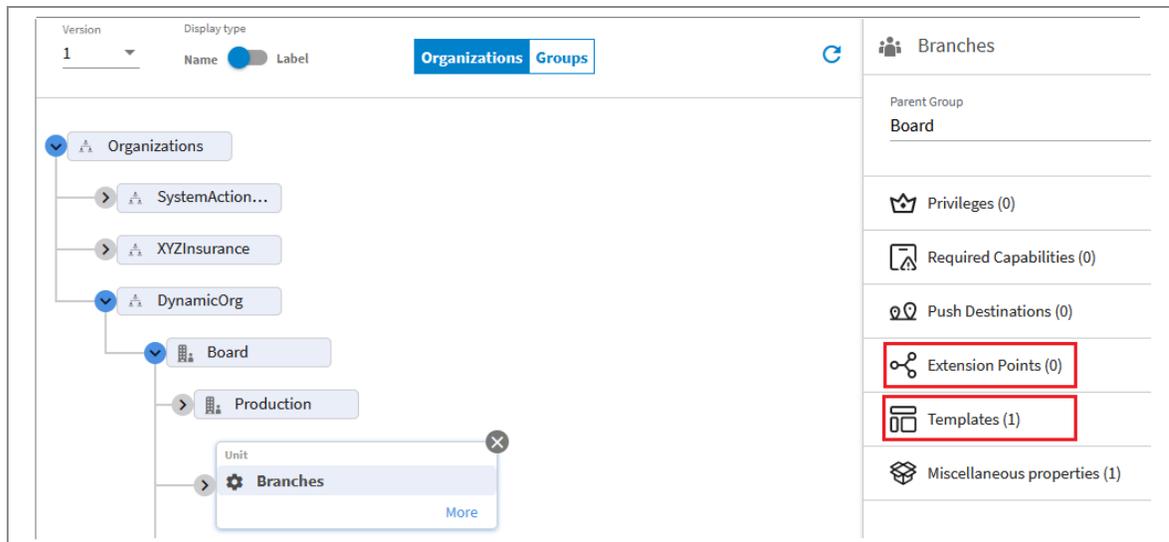
Before you begin

There must be an organization unit designated with an extension point, as well as an organization model template defined, in the deployed organization model.

Procedure

1. From the Organization Browser, click **Organizations**, expand the hierarchy and select the organization unit for which you want to configure an extension point, then click **More**.

The organizational unit must have been designated for an extension point when it was defined in TIBCO Business Studio - BPM Edition. When you select a dynamic organization unit, it includes **Extension Points** and **Templates** properties. For example:



2. Select the **Extension Points** property, then click **Configure Extension Point**.
3. Configure the extension point as follows:

Field	Description
Ldap Connection	The alias for the LDAP connection that contains the instance name attribute (see below) whose value is used to instantiate the organization model template.
Base dn	The LDAP branch to which the LDAP query (see below) will be restricted. This is optional and is relative to any Base-DN already specified on the LDAP connection.
Query	This expression will locate entries that identify the new dynamic organization model instances.
Search Scope	<p>Determines the depth to which the search will be performed, as follows:</p> <ul style="list-style-type: none"> • One Level - Only the elements directly within the Base-DN level are searched. • Sub Tree - Elements directly within, and below, the Base-DN level are searched.
Instance Name Attribute	For every LDAP entry that is found in the query result set, the query looks up this attribute. For every value in this attribute, an instance of the dynamic organization model is created. And each value in this attribute is used as the name of the root organization unit of the newly created dynamic organization model instance.
Dynamic Organization Identifier(s)	<p>These are attributes that are defined in the dynamic organization model. They are used to uniquely identify a generated instance of a dynamic organization at runtime. When a participant is assigned to a user task, the runtime needs to be able to identify the correct instance of the dynamic organization. These identifiers are used for that purpose.</p> <p>These attributes must be mapped to LDAP attributes that contain values used to identify a particular instance of the dynamic organization at runtime. This allows the process to access the identifying value (as processes cannot directly access LDAP</p>

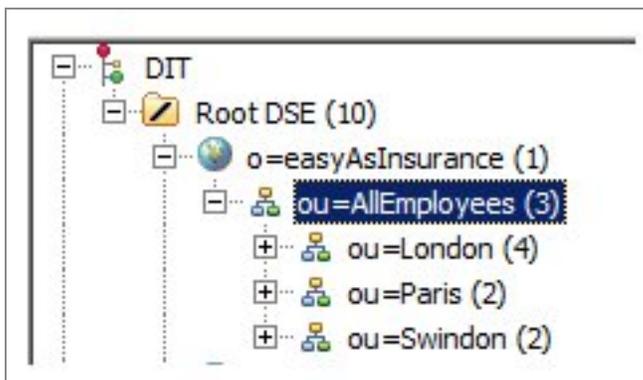
Field	Description
	<p>attributes).</p> <p>For each identifier listed, select an LDAP attribute that contains the values needed by the process to identify the dynamic organization model instance.</p> <p>For more information and to see examples of dynamic organization identifiers, see the "Dynamic Organization Identifier Mapping" topic in the <i>TIBCO Business Studio - BPM Edition Application Designer's Guide</i>.</p>

4. Click **Save** to save the extension point configuration.

Extension Point Configuration and Model Template Instance Generation Example

Assumptions:

- There is a dynamic organization model that was previously defined in TIBCO Business Studio - BPM Edition , with an extension point entitled "Branches":
- The organization model has been deployed, so it appears in the Organization Browser as shown in the example above:
- Your LDAP source looks like this:



- You want a branch dynamically generated for each of the organization units under `ou=AllEmployees` in the LDAP source.

- Your business process contains a dynamic organization identifier called "Town" that is mapped to the "ou" attribute, which allows work items to be routed to users in the appropriate dynamically generated branch.

To configure an extension point for Branches, follow the procedure above and fill in the fields of the Dynamic Organization Configuration dialog box as follows:

Add Extension Point

Ldap Connection *
easyAs

Base dn
ou=AllEmployees

Query
(objectClass=organizationalUnit)

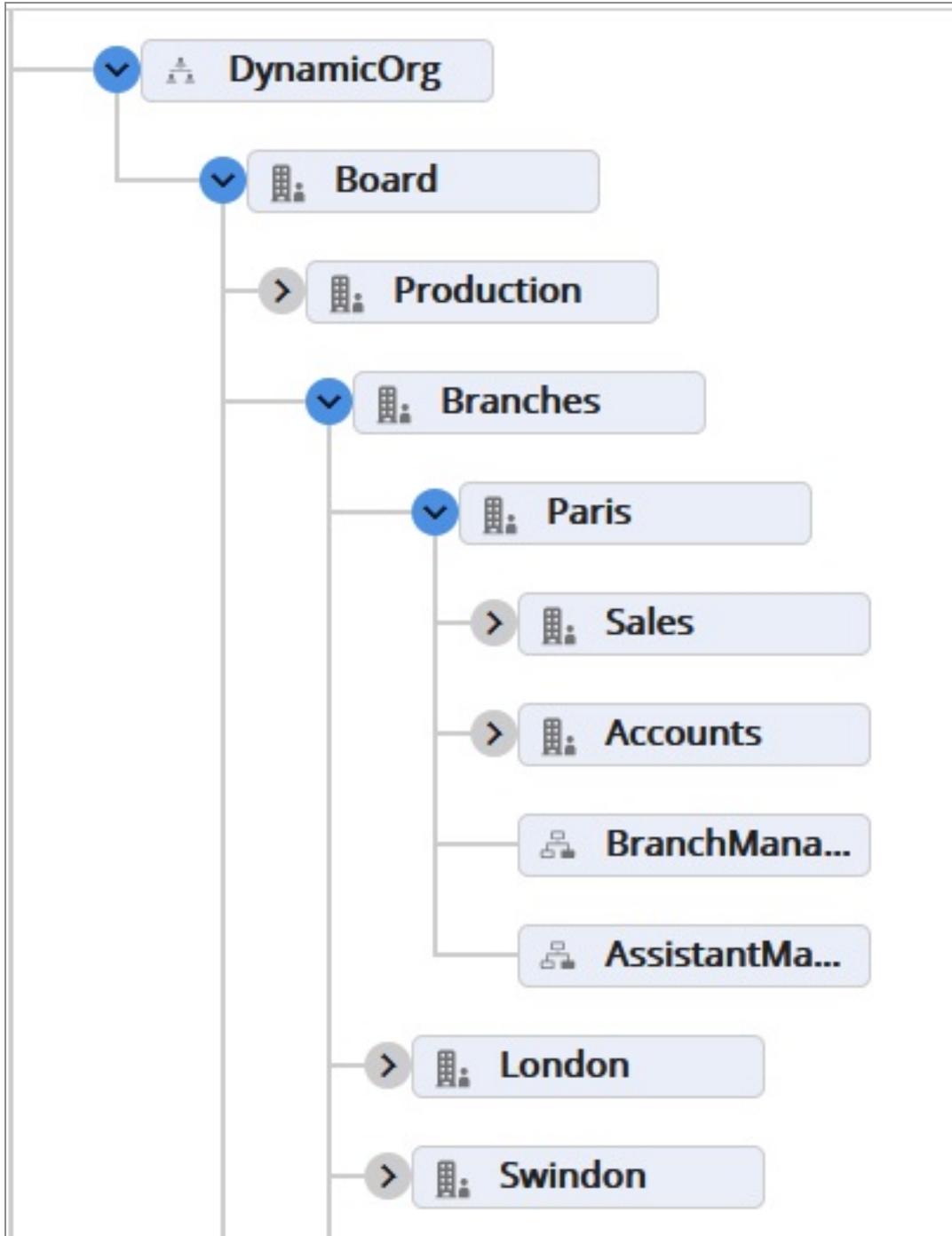
Search scope
 One Level Sub Tree

Instance Name Attribute *
ou

Dynamic Organization Identifier(s)
town

ou

After instances of the dynamic organization model are generated according to the [Directory Engine Configuration](#) properties, the organization model appears as follows in the Organization Browser:



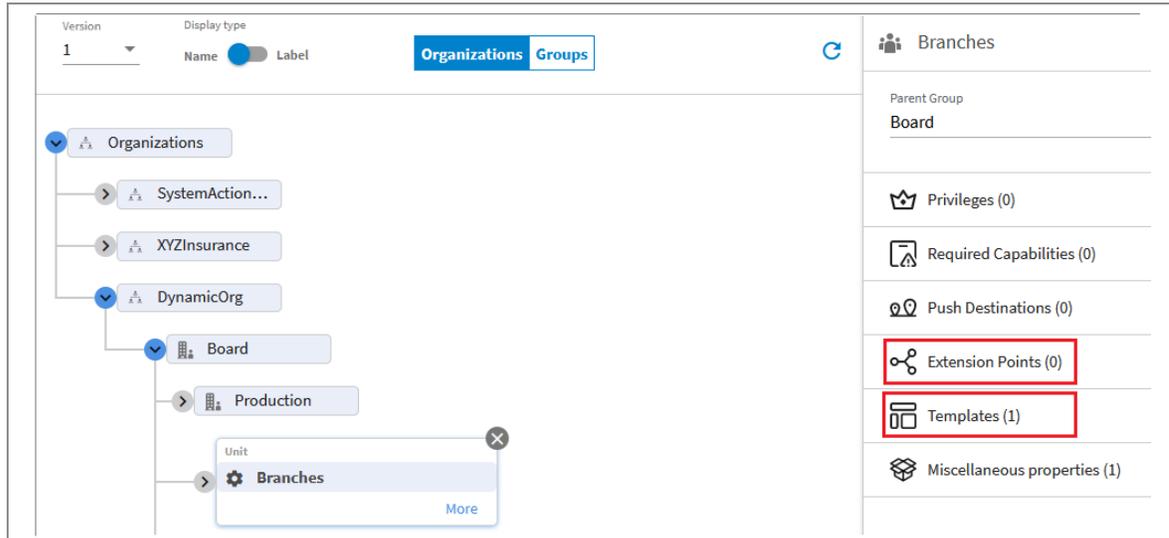
Viewing a Dynamic Organization Model Template

You can see the structure of dynamic organization model templates from the Organization Browser.

Procedure

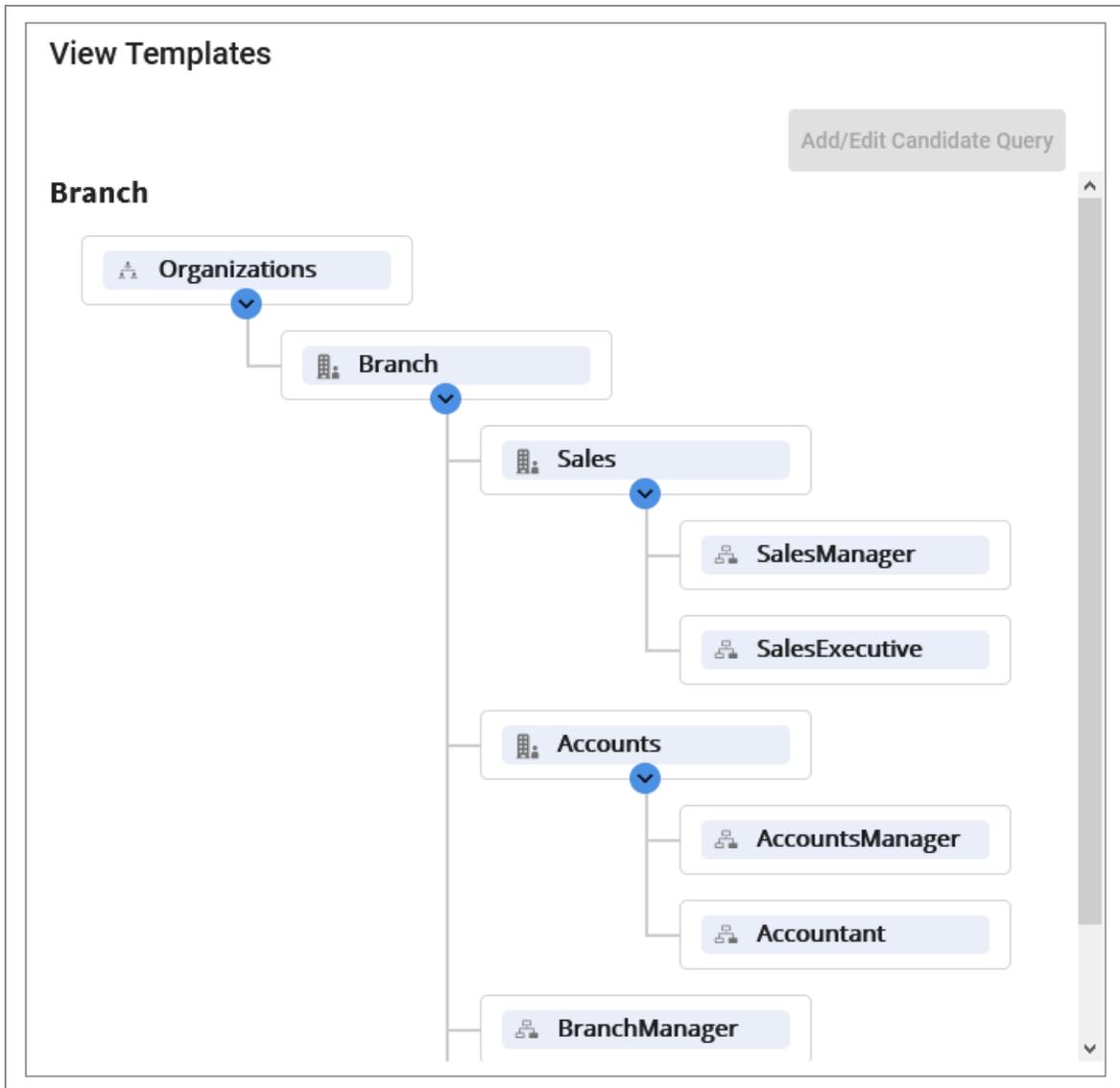
1. From the Organization Browser, click **Organizations**, then expand the hierarchy and select the organization unit for which you want to view the dynamic organization model template.

The organization unit must have been designated for an extension point when it was defined in TIBCO Business Studio - BPM Edition . When you select a dynamic organization unit, it includes **Extension Points** and **Templates** properties. For example:



2. Select the **Templates** property, then click **View Template**.

The structure of the dynamic organization is shown. For example:



Note: You can also define a candidate query from this dialog, using the **Add/Edit Candidate Query** button, to populate positions of the dynamically generated organization model. For more information, see [Populating Dynamic Organization Models](#).

3. Click **Close** to close the Dynamic Org Template dialog.

Populating Dynamic Organization Models

Organization models that are generated dynamically, can also be populated dynamically.

This allows you to generate organization units and their positions, and at the same time populate the positions in each instance of the organization unit with only the resources that are appropriate for each instance.

For information about defining candidate queries using substitution variables, see [Configuring Candidate Queries for Dynamic Organizations](#).

Candidate Queries

A candidate query is an LDAP Query assignment to a position or group. The position or group is populated based on the results of the candidate query.

An LDAP container must be specified in the candidate query configuration. The primary LDAP source of the LDAP container identifies the LDAP connection on which the query is performed. This also determines the LDAP container to which any newly created resources are assigned.

Any resource identified by the candidate query of a position or group must also be visible via the associated LDAP container. That is, no resource can be created dynamically that could not also be created manually using an LDAP container. This ensures that any resource attributes can retrieve their values from the mapped LDAP attributes of an LDAP container.

Each candidate query will only identify potential entries from the primary LDAP source of the associated LDAP container. If that LDAP container has any secondary LDAP sources, the rules that bind entries within the secondary LDAP sources to those of the primary LDAP source must be followed. It is only when those rules have been completed that the true set of candidate resources can be resolved.

The deletion of the LDAP container causes the deletion of all resources belonging to that LDAP container; whether they were created manually or dynamically. The deletion of the LDAP container always results in the deletion of candidate queries that reference that LDAP container.

Candidate queries can be used to populate either *static* or *dynamic* organization models:

- **Static Organization Models** - These are organization models that are statically defined in TIBCO Business Studio - BPM Edition, that is, they are not dynamically generated from model templates. For these types of organization models, you can populate both groups and positions using candidate queries.
- **Dynamic Organization Models** - These are organization models that are dynamically

generated from model templates. They consist of organization units with subordinate positions that can be populated using candidate queries. For dynamic organization models, the candidate query can also use *substitution variables* to identify the appropriate resources to assign to positions in each instance of the dynamically generated organization unit. This allows each instance to contain resources that are different than the other instances. (If you did *not* use substitution variables when assigning resources to a dynamically generated organization model, each of the instances would be populated with the same resources.)

LDAP Source Classes

The class of the primary LDAP source of the LDAP container determines how much candidate query configuration is allowed. There are two classes of LDAP sources:

- **LDAP Group Source** - For this LDAP source class, the candidate query will take *all* of the resources identified by the LDAP container as its candidate list. No configuration other than identifying the LDAP container is allowed for this class of LDAP source. So, it does not apply to populating dynamic organization models, that is, since neither a Base-DN nor a query is specified for this class of LDAP source, substitution variables cannot be specified.
- **LDAP Query Source** - For this LDAP source class, the candidate query can include a Base-DN and query to identify the resources to populate positions and groups.

Invoking Candidate Queries

Candidate queries are invoked using the following Directory Engine properties:

- `AutoResourceGenEnable` - Enables (true) or disables (false) the population of positions and groups that have candidate queries defined each day at the time specified in the `AutoResourceGenStart` property.
- `AutoResourceGenStart` - Specifies the time each day to populate positions and groups that have candidate queries defined, as long as `AutoResourceGenEnable` is set to true.
- `AutoResourceGenInterval` - The delay between the start of one candidate query processing event and the next. This value should be great enough to ensure that the two events do not overlap. The value is expressed as an XML Schema Duration string.
- `AutoResourceDeleteEnabled` - Enables (true) or disables (false) the removal of resources from positions / group if the LDAP entry/attribute from which they were

derived has been removed from the LDAP source.

For more information about Directory Engine properties, see [Directory Engine Configuration](#).

Configuring Candidate Queries for Static Organizations

This procedure describes how to configure candidate queries for *static* organizations, that is, organizations that are not *dynamically* generated.

Before you begin

You must have an LDAP container defined from which the resource candidates can be obtained.

Procedure

1. From the Organization Browser, click one of the following:
 - **Groups** button to configure a candidate query for a group.
 - **Organizations** button to configure a candidate query for a position.
2. From either the groups or organizations list, select the desired group or position, then click **More**.
3. In the right pane, select **Candidate Query**.
4. If there is already an existing candidate query configured, you can edit it by clicking , or delete it by clicking .
5. If there is no existing candidate query, click **Add Candidate Query** to add a new one.
6. Configure the candidate query using the fields on the **Candidate Query** dialog box, as follows:

Field	Description
Ldap	Select the LDAP container from which the resource candidates are to be

Field	Description
Container	<p>obtained to populate the group or position.</p> <ul style="list-style-type: none"> • If you select an LDAP container that was created using a query source, you can use the remainder of the fields on the Candidate Query dialog box to configure the candidate query. • If you select an LDAP container that was created using a group source, the remainder of the fields on the Candidate Query dialog box are disabled. In this case, the candidate query will take all the entries identified by the LDAP container (that is, the LDAP group) as its candidate list.
Base DN	<p>The LDAP branch to which the query (see below) will be restricted. This is optional and is relative to any Base-DN already configured on the primary LDAP source of the identified LDAP container. For example,</p> <p>ou=London</p> <p>If a Base-DN was configured on the primary LDAP source, it is shown below the Base DN field.</p>
Query	<p>This expression will locate entries that identify candidate resources. The expression is combined with that of the primary LDAP source of the identified LDAP container. The query expression must be enclosed in parentheses. For example,</p> <p>(employee=Permanent)</p> <p>The LDAP query that was specified on the primary LDAP source of the container is shown below the Query field.</p>
Search Scope	<p>Determines the depth to which the search will be performed, as follows:</p> <ul style="list-style-type: none"> • One Level - Only the elements directly within the Base-DN level are searched. • SubTree - Elements directly within, and below, the Base-DN level are searched. <p>The candidate query cannot be more inclusive in its Search Scope than</p>

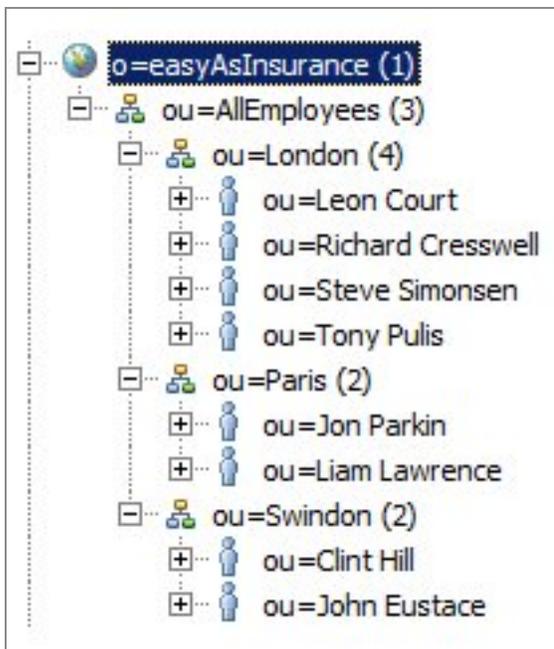
Field	Description
	the LDAP container's primary LDAP source. Therefore, if the primary LDAP source has a search scope of One Level, then the candidate query must also use One Level. However, if the primary source is SubTree, then the candidate query may be either.

- Click **Update** or **Add** to save the edited or new candidate query.

Candidate Query Configuration for a Static Organization Example

Assumptions:

- Your LDAP source looks like this:

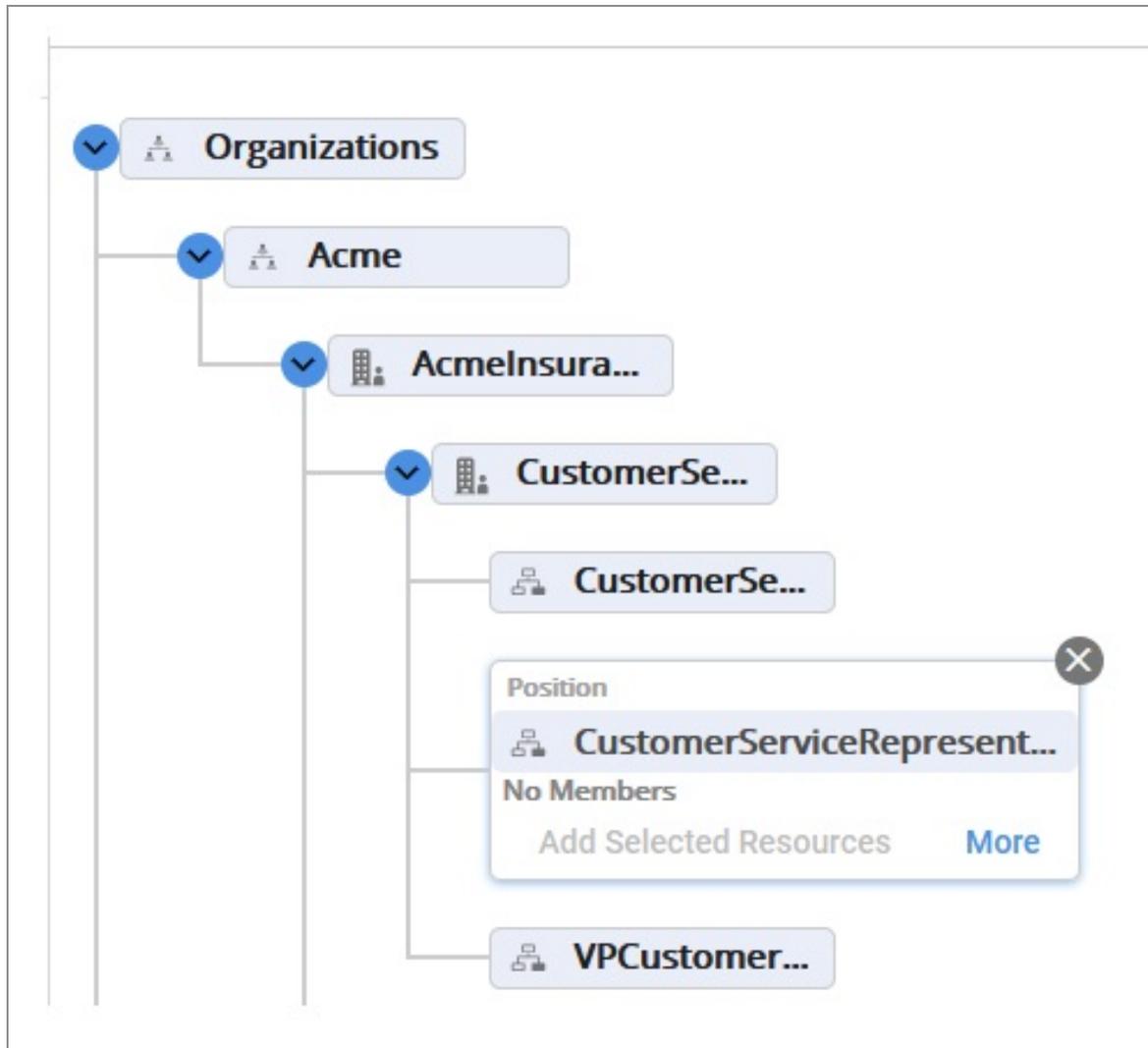


- The LDAP source contains the following attributes for each of the resources:

DN: ou=Richard Cresswell,ou=London,ou=AllEmployees,o=easyAsInsurance	
Attribute Description	Value
objectClass	<i>inetOrgPerson (structural)</i>
objectClass	<i>organizationalPerson (structural)</i>
objectClass	<i>person (structural)</i>
objectClass	<i>top (abstract)</i>
cn	Mr Richard Cresswell
sn	Cresswell
carlicense	Suspended
departmentnumber	FNB1
employeenumber	1320
employeetype	Permanent
givenname	Richard
mail	RCresswell@easyasinsurance.com
manager	ou=Tony Pulis,ou=London,ou=AllEmployees,o=easyAsInsurance
ou	Richard Cresswell
postaladdress	4 Cherry Walk, Mayfair, LONDON, EC1V
preferredlanguage	English, Welsh

The query in this example assigns candidate resources whose employeetype attribute = "Permanent".

- Your organization model contains a "Customer Service Representative" position to which you want the candidate query to assign candidate resources:



To configure a candidate query to populate the Customer Service Representative position with resources from the London office (`ou=London`), who are permanent employees (`employeetype=Permanent`), follow the procedure above and fill in the fields of the **Candidate Query** dialog box as follows:

Add Candidate Query

Ldap Container *

West ▼

Base Dn

ou=London,ou=Allemployees,o=easyAsInsurance

Query

(employeetype=Permanent)

Search scope

One Level Sub Tree

Cancel
Add

After the candidate query is invoked as configured by the Directory Engine properties, the position will be populated with resources that match the query.

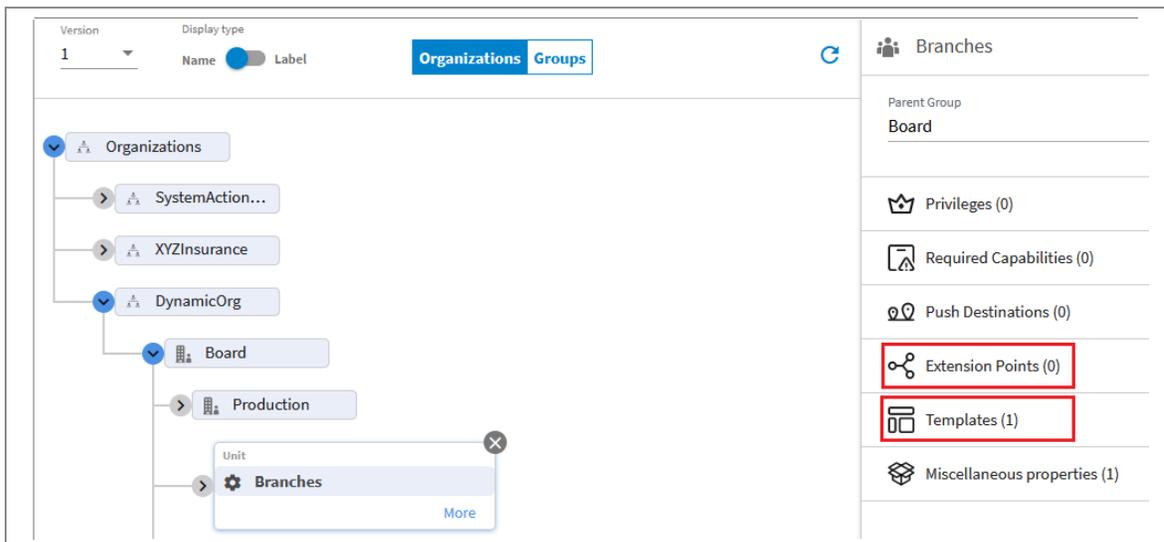
i Note: The deletion of a candidate query defined for a position or group unmaps all existing mappings to that particular position or group.

Configuring Candidate Queries for Dynamic Organizations

It is possible to construct the Base-DN and LDAP query of a candidate query in such a way that it identifies different resources for each instance of the dynamic organization model.

Before you begin

- You must have an LDAP container defined from which the resource candidates can be obtained.
- There must be an organizational unit that has been designated for an extension point when it was defined in TIBCO Business Studio - BPM Edition, that is, a model template has been defined for the organization unit. When you select a dynamic organization unit, it includes **Extension Points** and **Templates** properties (these properties are not shown for non-dynamic organization units). For example:



Substitution Variables

To allow you to specify that each instance of the dynamically generated organization unit is populated with different resources, two *substitution variables* are available for use in a candidate query:

- **{root-dn}** - The DN of the LDAP entry that initiated the organization model template instance. Generally, this substitution variable is used in the **Base DN** of the candidate query configuration. This is used in the example that is shown below.
- **{root-name}** - The name assigned to the root organization unit of the organization model instance; that is, the value of the LDAP attribute named in the extension point. Generally, this substitution variable is used in the **Query** of the candidate query configuration. Note that this variable is *not* used in the example that is shown below, but it is available for use if your LDAP source is set up in such a way that it needs to be queried.

Note that substitution variables must be enclosed in braces { }, and they are case insensitive.

These substitution variables will take their values according to each instance from the LDAP entry that the instance comes from. Using the following example, the {root-dn} for the currently selected LDAP entry is "ou=London,ou=AllEmployees,o=easyAsInsurance". The Paris and Swindon entries would have similar DN's. The {root-name} for the three organization units in this example are London, Paris, and Swindon.

The screenshot shows the LDAP Browser interface. On the left, the Project Explorer displays a hierarchy of organization units (OUs) under the Root DSE. The selected entry is 'ou=London', which is highlighted with a red box. Other OUs shown include 'ou=Paris' and 'ou=Swindon', also highlighted with red boxes. On the right, the Properties pane shows the DN for the selected entry: 'DN: ou=London,ou=AllEmployees,o=easyAsInsurance'. Below this, a table lists the attributes and their values for the selected entry.

Attribute Description	Value
<i>objectClass</i>	<i>organizationalUnit (structural)</i>
<i>objectClass</i>	<i>top (abstract)</i>
ou	London

Procedure

1. From the Organization Browser, click **Organizations**, expand the hierarchy and select the organization unit that contains an extension point, then select the **Templates** property.
2. Click **View Template**.
3. On the View Templates dialog box, expand the hierarchy to view the organization structure, select the position you want to populate in each dynamically generated instance of the template, then click **Add Candidate Query** (or **Edit Candidate Query** if you are editing an existing candidate query).
 - If there is already a candidate query defined for the selected position, it is displayed. You can use the information in the table below to edit the candidate query, or you can delete the existing candidate query by clicking the **Delete Query** () button.
 - If there has not been a candidate query defined for the selected position, click

Add Candidate Query then use the information in the table below to define the candidate query.

4. Configure the candidate query using the fields on the Candidate Query dialog box, as follows:

Field	Description
LDAP Container	<p>Select the LDAP container from which the resource candidates are to be obtained to populate the dynamic organization model.</p> <p>Note that if you choose an LDAP container that was created using a group source, all of the other fields on this dialog box are disabled. If you choose that type of container, you cannot use variable substitution in the Base-DN nor the Query, as those are disabled. In this case, every instance of the dynamically generated organization model will be the same (as defined in the LDAP container), and will be populated with the same resources.</p>
Base DN	<p>The LDAP branch to which the query will be restricted. This is optional and is relative to any Base-DN already configured on the primary LDAP source of the identified LDAP container.</p>
Query	<p>This expression will locate entries that identify candidate resources. The expression is combined with that of the primary LDAP source of the identified LDAP container. The query expression must be enclosed in parentheses.</p> <p>If you want to include multiple attributes in the query, they must be ANDed together using the following notation:</p> <p><code>(&(attribute1=value)(attribute2=value))</code></p> <p>For example:</p> <pre style="background-color: #e6f2ff; padding: 10px; border: 1px solid #d9e1f2;">(&(employeetype=Contract)(departmentnumber=3100))</pre> <p>You could include the root-name substitution variable in the query as well if it works with the data in the LDAP source. For example:</p>

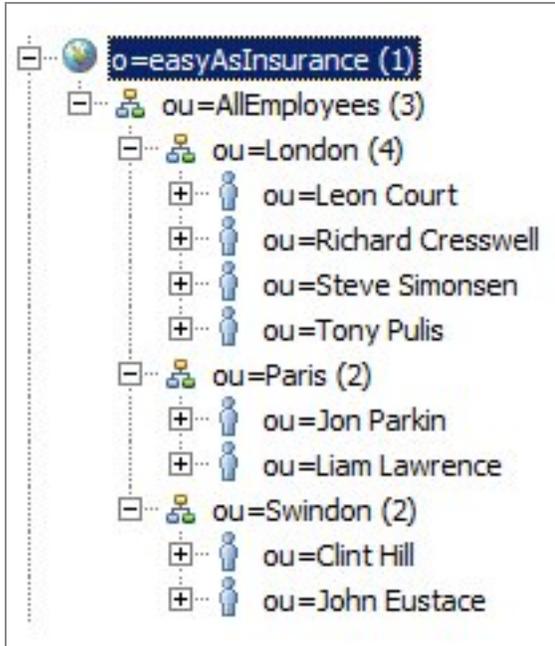
Field	Description
	<pre data-bbox="492 296 1419 415">(&(ou={root-name})(employee=Contract) (departmentnumber=3100))</pre>
Search Scope	<p data-bbox="492 457 1419 489">Determines the depth to which the search will be performed, as follows:</p> <ul data-bbox="537 516 1419 684" style="list-style-type: none"> <li data-bbox="537 516 1419 583">• One Level - Only the elements directly within the Base-DN level are searched. <li data-bbox="537 611 1419 684">• SubTree - Elements directly within, and below, the Base-DN level are searched. <p data-bbox="492 716 1419 905">The candidate query cannot be more inclusive in its Search Scope than the LDAP container's primary LDAP source. Therefore, if the primary LDAP source has a search scope of One Level, then the candidate query must also use One Level. However, if the primary source is SubTree, then the candidate query may be either.</p>

5. Click **Save Changes** to save the edited/new candidate query and close the Candidate Query dialog box.
6. Click **Close** to close the Dynamic Org Template dialog box.

Candidate Query Configuration for a Dynamic Organization Example

Assumptions:

- Your LDAP source looks like this:



- The LDAP source contains a departmentnumber for each of the individuals in the LDAP:

Complete Query Source Sample Details

givenname	sn	userpassword	ou	departmentn...	employeetype
Clint	Hill	dGliY28xMjM=	Clint Hill	FNB3	Permanent
John	Eustace	dGliY28xMjM=	John Eustace	FNB3	Temporary
Jon	Parkin	dGliY28xMjM=	Jon Parkin	FNB2	Permanent
Leon	Court	dGliY28xMjM=	Leon Court	FNB2	Permanent
Liam Lawrence	Lawrence	dGliY28xMjM=	Liam Lawrence	FNB1	Contract
Richard	Cresswell	dGliY28xMjM=	Richard Cress...	FNB1	Permanent
Steve	Simonsen	dGliY28xMjM=	Steve Simonsen	FNB1	Permanent

Each LDAP entry contains a `jobtitle` attribute. The query looks for the entries in which the value of this attribute is "SalesRep".

- You have configured the extension point for the dynamic organization as follows:

Add Extension Point

Ldap Connection *
easyAs

Base dn
ou=AllEmployees

Query
(objectClass=organizationalUnit)

Search scope
 One Level Sub Tree

Instance Name Attribute *
ou

Dynamic Organization Identifier(s)
town

ou

Note that this is the same extension point configuration described in [Configuring Dynamic Organization Model Extension Points](#).

- The candidate query for the "departmentnumber" in the model template is configured as follows:

Add Candidate Query

Ldap Container *

West ▼

Base Dn

{root-dn}

Query

(departmentnumber=FNB1)

Ldap Container Primary Source Query : (objectClass=person)

Search scope

One Level Sub Tree

Cancel
Update

After the candidate query is invoked, as specified by the Directory Engine [Invoking Candidate Queries](#), each instance of the dynamically generated position is populated according to the query.

Modifying Existing Candidate Queries for Dynamic Organizations

After an organizational unit is dynamically generated, and positions are dynamically populated based on the candidate query, you can modify the candidate query.

When you modify an existing candidate query, you can modify it in one of the following ways:

- **Modify the candidate query in the organization model template** - When you modify a candidate query at this level, the next time positions are generated (based on the property settings in the Directory Engine [Invoking Candidate Queries](#)), those positions will be populated based on the modified candidate query, except for those positions whose individual candidate queries have been modified (as described below).
- **Modify the candidate query for a generated position** - When a generated position's candidate query is modified, that specific candidate query is used to populate that position; the candidate query in the template is no longer used to populate the position. Therefore, if the candidate query in the template is modified, it does not impact positions whose individual candidate queries have been modified.

To modify the candidate query for either the template or a generated position, perform the procedure provided in [Configuring Candidate Queries for Dynamic Organizations](#), except Step 2 and Step 3 as described below, depending on which candidate query you want to modify.

To modify the candidate query in the organization model template, do the following:

1. Access the organization unit that contains an extension point, click **Templates**, then click **View Template**.
2. On the View Templates dialog box, expand the hierarchy to view the organization structure, select the position whose candidate query you want to modify, then click **Edit Candidate Query**.
3. Continue with Step 4 in [Configuring Candidate Queries for Dynamic Organizations](#) to modify the query as desired.

To modify the candidate query for a generated position, do the following:

1. Expand the organization unit hierarchy to the generated position whose candidate query you want to modify, and select the position.
2. Click **Candidate Query**.

3. On the Candidate query available entry, click 
4. Continue with Step 4 in [Configuring Candidate Queries for Dynamic Organizations](#) to modify the query as desired.

Process Manager

Process Manager displays information about the process templates in your system, such as their name, package, and version number. You can also start instances of your process templates, and perform actions such as suspend, resume or cancel processes. Depending on your privileges, you can also migrate processes to different versions of the same process template, and administer halted processes.

Process Manager displays columns that show you information about the process instances in your list. Some columns are displayed by default - see [Process Manager Columns](#). However, you can customize what columns are displayed, see [Customizing Process Manager Columns](#).

If you have a large number of processes in your list, you can search for a specific process instance, see [Searching for Process Instances](#).

Process Manager Columns

The following table describes the columns that show you information about the process instances in your list.

Column Name	Description
Process Instance Id	A unique alphanumeric value identifying this particular process instance.
Process Name	The name of the process instance.
Process ID	A unique alphanumeric value identifying this particular process template.

Column Name	Description
State	Indicates the current status of the process instance.
Start Date	The date and time the state of the process instance changed to Active.
Created Date	The date and time the process instance was started.
Actions	Provides the actions you can perform, such as canceling, suspending, and resuming process instances.

Not all columns are displayed by default. You can select which columns are displayed by selecting . See [Customizing Process Manager Columns](#) for more information. Note that some columns may be empty, depending on the state of your process instances.

Customizing Process Manager Columns

You can customize your process list so that only certain columns are displayed, or you can change the order in which the columns are displayed.

Procedure

1. Select .
2. Select the columns that you want to display from the dropdown list. See [Process Manager Columns](#).
3. Click away from the list to save your changes. The process list is now displayed based on the columns that you chose.

Process Instance State

The state of a process instance determines what stage that it has reached in the process. A process instance can have various states. The **State** column in Process Manager indicates the process instance's current state.

You can also filter your process instances by state, by selecting the process template of the process instances you are interested in and selecting the state you want to filter the process instances on.

The following table describes the possible states of a process instance and their meanings:

State	Description
ACTIVE	This is the default state of a process instance that has been initiated but is not suspended or halted.
SUSPENDED	The process instance has been suspended.
HALTED	The process instance has halted.

* Sub-process instances can be canceled in a "cascading" fashion, which causes them to be canceled from the "bottom-up". This is done using the "cancelation event handler."

Searching for Process Instances

If you have a large number of process instances in your list, you can use the search feature to search for a specific process template instance.

Procedure

1. Select the process template for the process instances you want to find.
2. Select .

You can search on the following:

3. Select the desired filter option:
 - **Filter Criteria** - This option can be used to search for the following:

Name	Description	Data Type
Package	The name of the package that contains the process	Text

Name	Description	Data Type
Name	template you are interested in. This is specified in TIBCO Business Studio - BPM Edition at design-time.	
Process Name	The name of the process template. This is specified in TIBCO Business Studio - BPM Edition at design-time.	Text
Version	Identifies the version of the process template.	Numeric
Instances Started	Identifies the instances started before or after a given date and time.	DateTime

- **Process Instance Id** - Use this option to search on the Process Instance ID.
4. Choose the appropriate filter operator (Equals to, Less than, Greater than, After, or Before), depending on what you are searching for.

When searching for names, you can use the 'Less than' and 'Greater than' operators to find names that begin with characters that are alphabetically before or after the specified character. For example, specifying 'Less than', and specifying "D", finds names that begin with "A", "B", and "C". Case is significant, however. If a process name begins with "W", specifying 'Greater than' "t" does not find process instances. But specifying 'Greater than' "T" does find process instances. You can also use the * wildcard character when searching for a name using 'Equals to'. For example, 'Equals to' "W*" finds process instances for processes that begin with "W".

5. Enter the value of what you are searching for, then click **OK**.

The applied filters can be removed by clicking on RESET and then **OK**.

Filtering Process Templates

Filtering a list of process templates involves entering filter criteria so that only some of the process templates are shown in Process Manager, rather than all of them.

This allows you to display only the process templates you are interested in. For example, you may only be interested in process templates with a version number later than 2.0.

The Filter Templates by: The dialog box allows you to build a "filter expression" that is applied to all process templates. If the process template satisfies the filter expression (for example, the process template has a version number of 2.5 and the instance was started after July 12th), it is shown in the process template list.

Procedure

1. Select **Templates**.
2. Select .

You can filter on the following:

Name	Description	Data Type
Package Name	The name of the process template. This is specified in TIBCO Business Studio - BPM Edition at design-time.	Text
Process Name	The description of the process template. This is specified in TIBCO Business Studio - BPM Edition at design-time.	Text
Version	Identifies the version of the process template.	Numeric

3. For each column, use the value field to enter the value for the items you want to be returned. How you enter a value depends on the data type for the column you have selected.
4. Select **OK**.
5. To cancel or reset the filter, select  to display the Filter Templates by: dialog again.

Starting a Process Instance

When you start a process instance, one or more processes may be started, resulting in processes appearing in the list.

Starting an instance of a process results in initiating the first task of the process. This typically results in one or more work items being sent to one or more work lists.

The process instance is considered complete when the final task in the process is completed.

Procedure

1. Select the process template of which you want to start a process instance.
2. Select **START**.

Suspending a Process Instance

You can suspend activity in a process instance, which causes all work items that are associated with that process instance to also become suspended, and the process flow is halted at that point. Work items that are suspended cannot be worked on, for example, they cannot be opened, allocated, re-offered, or canceled.

To suspend a process instance, you must have the Resume/Suspend Process instance system action.



Note:

- if you have:
 - work items that have a timer event configured, or
 - timer events defined within the processand the timer event is reached while the process instance is suspended, then the timer event is still processed. However, you will not see this activity until the process instance has resumed. For example:
 - If the process is configured to move to the next task when the timer event has fired, the process moves to the next task when the process resumes.
 - If the task that has a timer event is set to withdraw on expiry, then the task is withdrawn when the process resumes.
- work items that are suspended become hidden in the work item list.

If a work item is already open when the process instance is suspended, the work item can still be canceled, closed, or submitted:

- If canceled, any changes made on the work item form are discarded and the work item is returned to the work item list with a SUSPENDED state.
- If closed, any changes made on the work item form are saved, and the work item is

returned to the work item list with a state of SUSPENDED.

- If submitted, any new work items that result from the process flow appear in the appropriate work item lists, but they will have a SUSPENDED state.

The suspended work items cannot be worked on until the process instance is resumed — see [Resuming a Suspended Process Instance](#).

Procedure

1. Select the process template of which you want to suspend a process instance.
2. Select a process instance.
3. From **Actions**,  **>Suspend**.

When a process instance is suspended, its status changes to SUSPENDED.

Resuming a Suspended Process Instance

Resuming a suspended process instance causes the process to flow as usual. Work items that were suspended because their process instance was suspended can now be opened and processed normally.

To resume a suspended process instance, you must have the Resume/Suspend Process instance system action.

From the process template list, select one or more suspended process instances that you want to resume, then from **Actions**, select  **> Resume**. When a process instance is resumed, its status changes to ACTIVE.

Canceling a Process Instance

You can cancel a process instance, which stops the process flow and deletes all work items that are associated with that process instance.

To cancel a process instance, you must have the Cancel/Purge Process Instance system action.

Work items that are associated with a canceled process instance are removed from the work item lists of the user to whom the work items were offered/allocated the next time their work item lists are refreshed.

If a user has a work item open when its associated process instance is canceled, a message is displayed when the user attempts to cancel, close, or submit the work item form informing the user that the process instance has been canceled.

Procedure

1. From the process instance list, select a process instance, then from **Actions**, select  > **Cancel**.

A dialog box is displayed that asks you to confirm that you want to cancel the selected process instances.

2. Select **OK** to confirm.

When a process instance is canceled, it is removed from the process instance list.

Fixing a Process Instance

When a process encounters a problem, such as, it cannot contact a remote server as part of a REST service activity, it may enter the HALTED state. In this state, the process flow cannot continue. You can fix a process in this state through the Process Manager.

Using this option, you can inspect and change the data associated with the failed instance and resume, retry, ignore, or cancel the halted process instance.

To fix a halted process instance, perform the following:

Procedure

1. From the process instance list, select the halted process instance, then from **Actions**, select  > Fix process.

A Halted Process Management / View Status page is opened that displays the failure status for the selected process instances.

2. Determine the cause of the failure and choose from the following options to progress each halted process instance:

Option	Description
Cancel instance	<p>Cancels the process instance and stops the process flow.</p> <p>When a process instance is canceled, it is removed from the process instance list.</p>
Inspect Data	<p>Inspects or changes data to repair a damaged process instance.</p> <p>If you update data for the process instance, click Retry to progress the process flow.</p>
Ignore & continue	<p>The failed task is skipped in the halted process instance and the process instance continues processing from the point in the process after the failed task. This option only makes sense if the process is capable of continuing successfully without the skipped task having been performed.</p>
Retry	<p>Retries the halted process instance to progress normally. This option is used if the reason for the failed task has been resolved.</p>

Process Instance Migration Overview

Process migration is the ability to migrate a long-running process from one version to another version of the same process template. Process migration can only occur at specific points in the process template. These points are called migration points. Migration points are automatically identified by TIBCO Business Studio - BPM Edition at design-time.

When you perform process migration, all active instances of the process are migrated. It is not possible to specify an individual active process instance to migrate.

You must specify the source and destination process templates to migrate from and to. The migration points, in other words, the tasks that are common to the source and destination process templates, are automatically displayed. The point at which the active process instances migrate depends on the point they have reached in the process. Migration occurs when the process instance is about to execute a task that has been scheduled for migration. If the task has already been offered to a user or executed, in other words, the

work item has already arrived in the work list, it is too late to be migrated and migration will take place at the next migration point specified or, if no other migration points have been specified, carry on with its existing process template version.

i Note: To perform process migration, you must log in as a user with a privilege that has the `Handle Process Migration` system action assigned to it. This is configured in the Organization Modeler in TIBCO Business Studio - BPM Edition. For information, see *TIBCO Business Studio - BPM Edition Application Designer's Guide*.

How to Migrate a Process Instance

If a process has been configured for migration in TIBCO Business Studio - BPM Edition , you can migrate it in **Process Views**.

Procedure

1. From the process template whose process instances you want to migrate select **⋮ > Migrate**.
2. From the **Migrate From:** panel, select the source process template that the active process instances are executing against.
3. From the **Migrate To:** panel, select the destination process template that you would like the process instance to migrate to. The destination migration points are displayed.
4. Select **Migrate**.

i Note: All audit and statistics information that refers to the current process still apply after migration.

Purging Process Instances

Process instances that are completed, canceled, or failed are automatically purged from the system. You can also purge process instances that have a state of Active, Halted, or Suspended.

Before you begin

You must be logged in as a user who has been assigned the system action, Bulk Cancel/Purge Process Instance, to purge process instances.

You can either purge all process instances for a selected process template, or selected process instances.

- To purge all instances of a process template, select the template from the **Process Templates** list in Process Manager, then select **⋮ > Purge**.
- To purge individual process instances, select multiple process instances in Process Manager, then click **Purge Selected Instances** (note that **Purge Selected Instances** does not appear until you select more than one process instance).

A confirmation dialog box is displayed as this action cannot be reversed.

Contribute Your App

Contributing your app to TIBCO BPM Enterprise causes the app to appear in the Work Manager UI.

Procedure

1. From Administrator, select **Contribute Your App**.

All published applications that you added in Application Development (AppDev) are listed.

2. Drag and drop your selected app from the left pane to the right pane.

You can now access the app from the **More Apps** component in Work Manager.

3. Provide a name for the application, then click **Save**.

The newly created application appears under the 'More Apps' component in the Work Manager UI.

The app can be removed from the right pane by clicking **Remove App**. On clicking **Remove App**, the app is removed from the right pane, but not from more apps in the work manager. Currently, you cannot delete contributed app from more apps in the work manager.

Configuration Management

Each of the individual components in TIBCO BPM Enterprise can be configured by setting property values in Administrator. These settings can be used for performance and tuning purposes.

TIBCO BPM Enterprise properties are configured using Configuration Management:

1. From TIBCO BPM Enterprise Administrator, select **Configuration Management**.

If any properties had previously been configured for a particular component via **Configuration Management**, the component is listed in the left pane of the **Configuration Management**. If a component is listed, you can select the component to see the properties that have been configured for that component via Configuration Management. Properties that apply to the component but are not listed use their system-defined defaults.

For properties that are listed, you can click to edit the current value, or to delete the current value. If you delete the current value, it reverts to its system-defined default.

2. Click **Add Property**.
3. In the **Property Group ID** field, select the component for which you want to configure a property.
4. In the **Property Name** field, specify the name of the property as shown in the table below.
5. In the **Property Value** field, specify the desired value of the property.
6. Click **Save**.

To see a list of configurable properties, click the respective component name:

Component	Description
Authentication Configuration	Manages authentication of TIBCO BPM Enterprise users.
Business Resource Management	Schedules and performs actions on work items, as well as filters and sorts work items to create work item lists.
Note: There are no configurable properties for this component.	

Component	Description
Case Data Management Configuration	Manages the flow of case data.
Directory Engine Configuration	Manages resources and organization models.
Event Collector	Collects and stores events that occur during runtime.
Logging (Event Publication Configuration)	Controls the list of profiles to publish.
Pageflow Engine Configuration	Controls the processing of pageflows, which present a sequence of forms to the user for a single work item.
Process Engine Configuration	Manages process templates and process instances.
Statistics Collector	Using events from the Event Collector, it generates process instance and work item statistics reports.
Note: There are no configurable properties for this component.	
Work Presentation Configuration	Performs operations on work items, such as opening, closing, canceling work items.

Authentication Configuration

The Authentication component manages the authentication of TIBCO BPM Enterprise users.

The configurable properties for the Authentication component are:

Property	Default	Description
ssoMode	<i>None</i>	Possible values are SAML or OpenId. Values are case insensitive.

Property	Default	Description
bpmSessionTimeout	5	The number of minutes for session timeout.
basicAuthEnabled	true	Specifies whether basic authentication is enabled. Values are true or false.

Case Data Management Configuration

The Case Data Management component manages the flow of case data.

The configurable properties for the Case Data Management component are:

Property	Default Value	Description
autopurge.interval	15	The interval, in minutes, between auto-purge cycles. The interval must be in the range 15 to 35791394. The minimum value of 15 minutes is to prevent it from being configured to run too frequently and creating an unnecessary load on the system. The maximum is <code>Integer.MAX_VALUE/60</code> , as we internally convert this to seconds and it has to fit in an Integer.
autopurge.purgeTime	129600	The number of minutes that must elapse after a case enters a terminal state before it qualifies for purging. This must be in the range 0 to 5256000, where 0 means to purge as soon as the case enters a terminal state, and 5256000 means to purge when the case has been in a terminal state for 10 years.
caseldsCacheSize	50	The cache size of the case id sequence. These caches store unique identifiers for various case types within the system. When a new case of the appropriate type is required, its ID can be assigned from the cache without needing a call to the

Property	Default Value	Description
		database. The size chosen for a cache can have an impact on system performance. For example, a cache size that is too large consumes unnecessary resources whereas a small cache size needs frequent calls to the database for a new batch of identifiers.

Directory Engine Configuration

The Directory Engine component manages resources and organization models.

The configurable properties for the Directory Engine component are:

Property Name	Default Value	Description
adminLdapAlias	system	
adminLdapName	tibco-admin	The name of the internal user who is authorized to log in until another user is configured. See Configuring the Admin User .
AutoResourceDeleteEnabled	false	Enables, or disables, the automatic deletion of those resources deemed to be invalid. A resource is deemed to be invalid when the LDAP entry, from which it originates, can no longer be found. When this property is true, the processing of invalid resources is performed as part of the dynamic population of organization models. This property only applies when AutoResourceGenEnable is true.
AutoResourceGenEnable	true	Enables, or disables, the processing of resource candidate queries; to automatically populate

Property Name	Default Value	Description
AutoResourceGenStart	03:30	<p>positions and groups with resources, according to configured LDAP queries.</p> <p>In a multi-node environment, only one node is chosen (at random) to schedule and process the candidate query processing events. If that node is stopped, another node will be chosen to take over the scheduling and processing.</p> <p>The time of day (expressed in the JVM's default time-zone) at which the first candidate query processing will be scheduled. The accepted format for this value is "hh:mm"; where "hh" is a value from "00" to "23", and "mm" if a value from "00" to "59".</p> <p>In a multi-node environment, only one node is chosen (at random) to schedule and process the candidate query processing events. If that node is stopped, another node will be chosen to take over the scheduling and processing.</p> <p>Subsequent processing events are scheduled according to the value of the property AutoResourceGenInterval.</p> <p>This property only applies when AutoResourceGenEnable is true.</p> <div data-bbox="824 1394 1414 1535" style="background-color: #f0f0f0; padding: 5px;"> <p>Note: Candidate query processing should be scheduled during off-peak hours when there is no user activity.</p> </div>
AutoResourceGenInterval	P1D	<p>The delay between the start of one candidate query processing event and the next. This value should be great enough to ensure that two events do not overlap. If a processing event does not complete within the specified interval,</p>

Property Name	Default Value	Description
		<p>events may be skipped.</p> <p>The value is expressed as an XML Schema Duration string (for example, "P1D" = once per day, "PT12H" = once every 12 hours).</p> <p>This property only applies when <code>AutoResourceGenEnable</code> is true.</p>

Event Collector

The Event Collector component collects and stores events that occur during run time. The component controls how and when the audit data is purged.

The configurable properties for the Event Collector component are:

Properties	Minimum Value	Maximum Value	Description
<code>purge.interval</code>	86400	1800 to 604800	The time (in seconds) between each run of the automated purge operation.
<code>purge.timeofday</code>	(not set = -1)	-1 to 86400	The approximate number of seconds after midnight (00:00) to start the purge. If the

Properties	Minimum Value	Maximum Value	Description
			value is set to -1 or the value is not set, then a random start time is used.
purge.batchinterval	120	0 to 300	The time (in seconds) between each batched purge of items. The items are purged in several small batches to reduce load and locking on the database.
purge.batchsize	1000	1 to MAXLONG	The number of root entries to delete on each batched purge attempt.
purge.retentionperiod	2592000	-1 to MAXLONG	The number of seconds for which a complete audit entry is retained. The

Properties	Minimum Value	Maximum Value	Description
			default period is 30 days.
purge.retentionperiod.casedeployment	purge.retentionperiod	-1 to MAXLONG	The number of seconds for which a complete case deployment audit entry is retained. The default setting is same as the global retention period setting.
purge.retentionperiod.case	purge.retentionperiod	-1 to MAXLONG	The number of seconds that a complete case audit entry is retained. The default setting is same as the global retention period setting.
purge.retentionperiod.user	-1	-1 to	The number

Properties	Minimum Value	Maximum Value	Description
		MAXLONG	of seconds for which a complete user audit entry is retained. The default setting is never to purge.
purge.retentionperiod.application	-1	-1 to MAXLONG	The number of seconds for which a complete application audit entry is retained. The default setting is never to purge.
purge.retentionperiod.process	purge.retentionperiod	-1 to MAXLONG	The number of seconds for which a complete process audit entry is retained. The default setting is same as the global retention period setting.

Event Publication Configuration

Event Publication table (ep_event) is a database table to which events are published, when the events are properly configured with a set of properties. A broker can then be produced as a microservice that reads and deletes events from this event publication table and publishes them, as required.

Schema

The schema for ep_event is:

```
CREATE SEQUENCE ep_event_seq START WITH 1 INCREMENT BY 1;

CREATE TABLE ep_event
(
  id          numeric(28)          DEFAULT nextval('ep_event_
seq'),
  format_version integer          DEFAULT 1,
  creation_time timestamp without time zone DEFAULT current_timestamp,
  details     text                NULL,
  CONSTRAINT pk_ep_event PRIMARY KEY (id)
)WITH (OIDS=FALSE);
```

The actual event data is provided in the details field as JSON. Check the following sample entry in the details column.

```
{
  "hostName": "bpm-ace",
  "messageCategory": "PROCESS_INSTANCE",
  "componentId": "BX",
  "creationTime": "2021-04-08T04:15:57.199+0000",
  "componentClassName": "com.tibco.bx.n2.ec.BxAuditTrail",
  "procId": 302,
  "moduleName": "SampleBPMPProject",
  "methodId": "processInstanceStateChange",
  "managedObjectVersion": "1.0.0.20200521145425675",
  "principalId": "tibco-admin",
  "simpleClassName": "BxAuditTrail",
  "uuid": "1cb15214_716e_4d0f_a602_cb999d390436",
  "threadId": 268,
  "parentContextId": "d9e8039e_4e92_4f13_bb6b_6b18bd650416",
  "parentObjectId": "p:0a218e",
```

```

"managedObjectName": "SampleBPMProjectProcess",
"rootProcessInstanceId": "",
"correlationId": "ca20227d_8919_40dc_a727_9a4fb4708213",
"applicationName": "ContainerEngineApp",
"severity": "AUDIT",
"processInstanceId": "409",
"procInstanceId": 409,
"methodName": "processInstanceStateChange",
"messageId": "BX_INSTANCE_PROCESS_STARTED",
"contextId": "549e08f5_94f6_4c3c_9a9d_1649b01043a8",
"eventType": "MESSAGE",
"message": "Process Instance started.",
"processPriority": "NORMAL",
"threadName": "ce_4",
"managedObjectId": "p:0a20bd",
"hostAddress": "172.18.0.4"
}

```

Configure Profile Property

Configure the `publishRulesConfig.profiles` property on the **logging** component to control what event should be published. It needs to be set with a comma-separated list that consists of one or more of the following values:

Property	Default	Description
<code>publishRulesConfig.profiles</code>	<i>None</i>	A list of possible values for the property include: PROCESS_INSTANCE, PROCESS_INSTANCE_TASKS, PROCESS_INSTANCE_EXCEPTIONS, CASE, WORK_ITEM, SYSTEM_LOGIN, SYSTEM_OTHER, STATS, DEPLOYMENT,

Property	Default	Description
		SYSTEM_CONFIG
		<p>Note: You can have more than one property value for event publication. In case of more than one property value, make sure that the values are separated by a comma.</p>

Logging

Logging refers to the recording of all events generated by TIBCO BPM Enterprise.

Logging data can be used for numerous purposes, ranging from debugging within a system, through to storage for non-repudiation logs, and all messaging in between.

This section explains the override property and format property.

Configure Override Property

Configure the following properties to modify the root level logger from INFO to the other log levels:

Property	Default	Description
logRulesConfig.root	INFO	Default log level used. The values can be TRACE, DEBUG, INFO, WARN, or ERROR.
logRulesConfig.overrides		Defines the overrides to apply to the base root severity when generating logging. Example: <code>com.tibco.n2=DEBUG, com.tibco.n2.brm=TRACE</code>

Configure Format Property

Configure the log outputFormat property to determine the message format:

Property	Default	Description
logOutputFormat	<i>Column</i>	Determines what format of message that you emit to the console (has no bearing on RSYSLOG that has to be RFC5424). The range value can be Enum of either RFC5424 or COLUMNS or JSON.

Pageflow Engine Configuration

The Pageflow Engine component controls the processing of pageflows, which present a sequence of forms to the user for a single work item.

The configurable property for the Pageflow Engine component is:

Property	Default	Description
procModuleMaxEntries	1000	The number of entries of the latest process modules in the cache.

Process Engine Configuration

The Process Engine component manages process templates and process instances.

The configurable properties for the Process Engine component are:

Property	Default	Description
BPEL Maintenance		
bpelPendingMessageProcessorNumMessages	500	The total number of

Property	Default	Description
		messages the background thread should process until it goes back to sleep for another interval.
bpelPendingMessageProcessorInterval	P30	<p>This property specifies how frequently a background job checks for, and purges, unclaimed messages that have past their expiration time.</p> <p>Specified as either of the following:</p> <p>P# - Where “P” indicates “periodic” and # is the number of minutes. For example, P60 causes it to check every 60 minutes. The lowest valid value is P1.</p> <p>D# - Where “D” indicates “daily” and # is the hour number (1-24). For example, D20 causes it to check daily at 8 PM.</p>
bpelPendingMessageProcessorBatchSize	50	The number of messages that the background thread should process in a single transaction.
bpelUnclaimedPendingMsgProcessorInterval	P30	<p>This property specifies how frequently a background job checks for, and processes, unclaimed messages.</p> <p>Specified as either of the following:</p>

Property	Default	Description
		<ul style="list-style-type: none"> • P# - Where "P" indicates "periodic" and # is the number of minutes. For example, P60 causes it to check every 60 minutes. The lowest valid value is P1. • D# - Where "D" indicates "daily" and # is the hour number (1-24). For example, D20 causes it to check daily at 8 PM.
bpelUnclaimedPendingMsgProcessorNumMsgs	500	This property controls the maximum number of unclaimed messages that will be picked up per execution. The minimum is 50.
bpelUnclaimedPendingMsgProcessorStart	false	Determines whether the background job to clear out unclaimed pending messages is run. For internal use only.
Global signals		
globalSignalProcessorInterval	P30	Specifies how frequently expired global signals are cleaned up. Specified as either of the following: <ul style="list-style-type: none"> • P# - Where "P" indicates "periodic"

Property	Default	Description
		<p>and # is the number of minutes. For example, P60 causes the cleanup job to run every 60 minutes.</p> <ul style="list-style-type: none"> • D# - Where "D" indicates "daily" and # is the hour number (1-24). For example, D20 causes the cleanup job to run daily at 8 PM.
globalSignalProcessorNumMessages	500	The batch size for expired global-signal cleanup jobs, that is, the number of expired signals that are to be processed at one time.

Work Presentation Configuration

The Work Presentation component performs operations on work items, such as opening, closing, canceling, and so on.

The configurable properties for the Work Presentation component are:

Property	Default	Description
formRedirectURL	https://localhost:8080/bpm/	The base URL where form artifacts are deployed.
smtpConnection	<i>None</i>	Name of the SMTP shared resource.

Authentication

An authenticated user is required to access TIBCO BPM Enterprise. Users must be registered with the TIBCO BPM Enterprise Directory Engine via the Organization Browser.

TIBCO BPM Enterprise supports the following types of authentication:

- **Basic Authentication**- The credentials used for authentication are obtained from the HTTP request in the form of a user name and password. The user name and password are authenticated against an LDAP.
- **SAML Web Profile** - If your TIBCO BPM Enterprise application is configured to use SAML Web Profile for authentication, users of your application can log in using a user name and password issued by an Identity Provider (IdP) that supports SAML Web Profile.
- **OpenID Connect** - If your TIBCO BPM Enterprise application is configured to use OpenID Connect, the users can log in with a user name and password issued by an Identity Provider (IdP) that supports OpenID Connect.

Authentication Process

TIBCO BPM Enterprise contains a login module for each of the available types of authentication; basic, SAML Web Profile, and OpenID Connect. When a TIBCO BPM Enterprise HTTP endpoint is accessed, the appropriate login module handles the user authentication by performing the following steps:

1. The system checks for a current user session, and whether or not it has expired. If a current user is in session, the HTTP request is processed.
2. If there is no current user session, a check is made to determine if TIBCO BPM Enterprise is configured for basic authentication. Basic authentication is HTTP basic authentication. In HTTP basic authentication, the principal's credentials are passed in the HTTP Authorization request header.

The basic authentication login module extracts the principal from the HTTP authorize header (if it is available) and searches for the user in TIBCO BPM Enterprise system. If the user exists in TIBCO BPM Enterprise, the system returns details of the user,

including the primary LDAP to be used for authentication purposes.

Basic authentication is configured using an HTTP Client Shared Resource defined in TIBCO BPM Enterprise Administrator.

3. If basic authentication is not used or fails, the system checks if TIBCO BPM Enterprise is configured for Single Sign-On (SSO) authentication (SAML Web Profile or OpenID Connect). SSO authentication must be configured if a basic authentication is not configured. Also, only one of the SSO authentication type configurations is supported across all in-bound TIBCO BPM Enterprise REST APIs at a given time (although, both types can be configured, only one can be enabled at a time).

Depending on which SSO authentication type is configured, control is handed over to the appropriate login module (SAML Web Profile or OpenID Connect), which uses the appropriate shared resource configuration defined in TIBCO BPM Enterprise Administrator.

After SSO authentication is completed, an authorization check is performed to ensure that the user exists in TIBCO BPM Enterprise. This is done by looking up the user in Directory Engine. If this is successful, the user is considered as authentic and an HTTP session is created.

SAML Web Profile Authentication

If your TIBCO BPM Enterprise application is configured to use SAML Web Profile for authentication, users can log in with a username and password issued by an IdP that supports SAML Web Profile. TIBCO BPM Enterprise supports Google and simpleSAMLphp SAML IdP.

 **Note:** Ensure that the resource registered with your IdP is added to the LDAP.

Perform the following procedure to ensure that SAML authentication works with your registered users:

1. Set up your preferred SAML Idp to download to your local machine. For more information, visit the website of your IdP provider.
2. Configure your SAML Idp. For more information about configuring a SAML shared resource, see [SAML Authentication Shared Resources](#).
3. Ensure that the user whose login credentials are registered with the Idp is also added

to the LDAP Container. For more information, see the Configure the LDAP Directory Server topic in the *TIBCO BPM Enterprise Installation Guide*.

The following steps describe the basic flow when a user attempts to log in to a TIBCO BPM Enterprise application, which is configured to use SAML Web Profile, using their IdP credentials. In this scenario, the user is not already logged in to TIBCO BPM Enterprise.

1. The user starts a TIBCO BPM Enterprise application that is using SAML Web Profile authentication.
2. The application tries to access the TIBCO BPM Enterprise server, but the login module determines that the user is not authenticated and that authentication is provided by SAML Web Profile.
3. The application redirects the login request to the IdP.
4. The IdP displays a login screen (for example, Google's login screen), requesting the user's IdP-issued credentials.
5. The user enters their IdP-issued credentials.
6. Upon receiving the user validation from the IdP, the application redirects the request to the TIBCO BPM Enterprise server to authenticate the user before logging the user in to the application.

A cookie is also created when the user is validated by the TIBCO BPM Enterprise server. The cookie is used to establish the session that is used by all subsequent calls to the TIBCO BPM Enterprise server.

The following steps describe the events that occur when an IdP-authenticated user logs out of a TIBCO BPM Enterprise application:

- The user is redirected to the login page for the application. When the request is redirected to `<domain>/apps/login/index.html`, the login page checks for an existing authenticated session. If there is no authenticated session, it forwards the request to the SAML IdP provider login page (if the user is not authenticated with the IdP).
- The cookie that was created upon login is removed.

OpenID Connect Authentication

If your TIBCO BPM Enterprise application is configured to use OpenID Connect, users of your application can log in using a username and password issued by an Identity Provider

(IdP) that supports OpenID Connect. TIBCO BPM Enterprise supports Google and PingID® OpenId IdP.

 **Note:** Ensure that the resource registered with your IdP is added to the LDAP.

Perform the following steps to ensure that OpenId authentication works with your registered users:

1. Set up your preferred OpenId IdP to download to your local machine. For more information, visit the website of your IdP provider.
2. Configure your OpenId IdP. For more information about configuring an OpenID authentication shared resource, see [OpenID Authentication Shared Resources](#).
3. Ensure that the user whose login credentials are registered with the IdP is also added to the LDAP container. For more information, see the Configure the LDAP Directory Server topic in the *TIBCO BPM Enterprise Installation Guide*.

The following steps describe the basic flow when someone attempts to log in to a TIBCO BPM Enterprise application, which is configured to use OpenID Connect, using their IdP credentials. In this scenario, the user is not already logged in to TIBCO BPM Enterprise.

1. The user starts a TIBCO BPM Enterprise application that is using OpenID Connect authentication.
2. The application tries to access the TIBCO BPM Enterprise server, but the login module determines that the user is not authenticated and that authentication is being provided by OpenID Connect.
3. The application redirects the login request to the IdP.
4. The IdP displays a login screen, requesting the user's IdP-issued credentials.
5. The user enters IdP-issued credentials.
6. After validating the user, the IdP returns an ID token in the form of a JSON Web Token (JWT) to indicate a successful authentication.

i Note: OpenID **Access Token** is not currently supported, but the login module determines that the user is not authenticated and that authentication is being provided. The OpenID **ID Token** is used to identify the user.

The response from the IdP also includes the *claims* specified in the **Auth Scope** field of the OpenID Authentication shared resource.

The IdP sends the ID Token and claims information to the "Redirect URI" that is specified in the OpenID Connect shared resource.

7. On receiving the ID token from the IdP, the application redirects the request to the TIBCO BPM Enterprise server to authenticate the user before logging in to the application.

A cookie is also created when the user is validated by the TIBCO BPM Enterprise server. The cookie includes the ID Token, which is used to establish the session that is used by all other subsequent calls to the TIBCO BPM Enterprise server.

The following steps describe the events that occur when an IdP-authenticated user logs out of a TIBCO BPM Enterprise application:

- The browser sends the value in the **Logout path** property to the TIBCO BPM Enterprise server. (When a user logs out of the TIBCO BPM Enterprise application, the user does not log out of the IdP but only invalidates the client session.)
- The cookie that was created on login, is removed.

i Note: At a given point, only a single SSO related shared resource can be enabled, SAML, or OpenID.

System Actions

System actions are predefined system tasks that users may wish to perform, but which an organization may wish to control access to.

System actions provide access to a wide range of functions - for example, work list and work item management, process management, and user administration. The following list contains a small sample of the available system actions.

User Admin

Administer users (resources) using the Organization Browser.

View Work List

View another user's work list.

Open Other Resources' Items

Open work items that are currently allocated to other users.

Open Work Item Audit Trail

Open the audit trail for a work item.

Each system action has a system-wide default value, which is either:

- Allowed - The system action can be performed by any user without authorization.
- Denied - The system action cannot be performed by any user unless they have the correct authorization.

This default value can be overridden for individual users by using *privileges*. In TIBCO Business Studio - BPM Edition, an analyst can assign privileges required to execute a particular system action against specific entities in the organization model. At runtime, only users who hold the required privileges will be able to perform that system action.

i **Note:** See [System Actions and Organization Model Versions](#) for the effect of changing the assignment of privileges between versions of the organization model.

If a set of required privileges for a given system action is assigned to an organizational entity, that setting will apply to all users assigned to that organizational entity. Any users not holding those required privileges will be denied access to that system action, for that organizational entity.

For a list of all available system actions, see "System Actions Reference" in the *TIBCO Business Studio™ - BPM Edition Application Designer's Guide*.

Scope of System Actions

The scope of a system action is defined by where in the organization model the privilege required to perform that system action is assigned:

- The default scope of a system action is system-wide. (Either any user or no user can perform the system action.)
- Privileges can be assigned to any system action at the organization model level.
- Privileges can also be assigned to some system actions at the level of the organizational unit, position, or group.
- A user can always perform certain actions (for example - View Work List and Set Resource Order Filter Criteria) if they are themselves the explicit scope of that action - that is, if they are not just related by position or group.

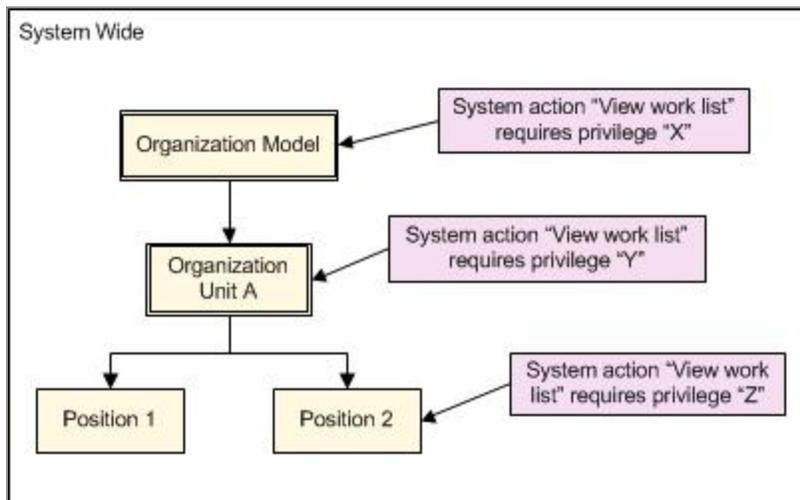
For example, the following table shows the default value and the possible scope for the system actions mentioned in the preceding section.

System Action	Permitted for all users by default?	Assign a privilege required to perform this system action to:	
		- an Organization Model?	- an Organization Unit, Position, or Group?
User Admin	Yes	Yes	No
View Work List	No	Yes	Yes
Open Other Resources' Items	No	Yes	Yes

System Action	Permitted for all users by default?	Assign a privilege required to perform this system action to:	
		- an Organization Model?	- an Organization Unit, Position, or Group?
Open Work Item Audit Trail	Yes	Yes	No

If different privileges are assigned to the same system action at different levels in the organization model, each level is checked when determining whether a user has the necessary privilege to be able to perform a particular system action. If the lowest level has no required privileges assigned, the parent entity is checked, and so on up the organization model hierarchy to the default, system-wide value.

For example, in the following diagram, the View Work List system action has been associated with three different privileges, "X", "Y" and "Z", at three different levels - the organization model, organization unit A, and position 2.



This means that a user must hold the privilege "X", "Y" or "Z" to view the work list of a user who holds Position 2.

If a user wants to view the work list of a user who holds Position 1, they must hold privilege "X" or "Y". This is because no privilege has been associated with Position 1, so any privileges associated with the parent entity are used instead. If privilege "Y" had not been associated with Organization Unit A, the user would instead need privilege "X", defined in the parent Organization Model.

As well as assigning different privileges at different levels, as shown above, qualifiers on the same privilege can be used to refine how access to a particular system action is controlled. (When comparing a required privilege to a held privilege, if either side is not qualified, the comparison is positive. If both sides are qualified, the qualifications must match for the comparison to be positive.)

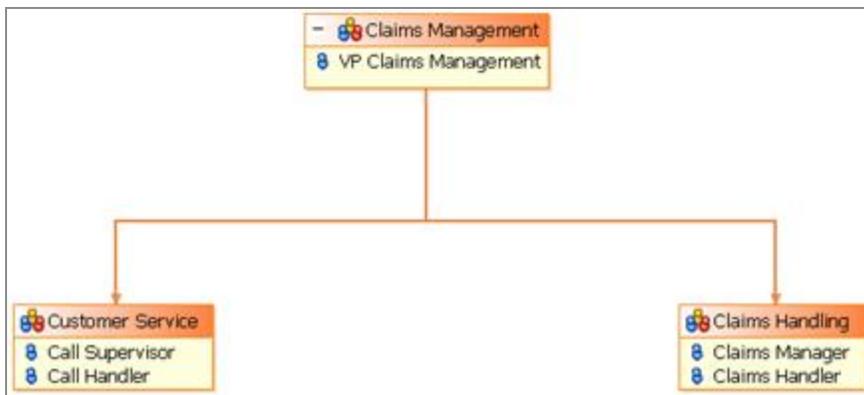
Controlling access to system actions by the application of (user-defined) privileges within the organization model provides an organization with a powerful and completely flexible way to customize and tailor users' access to system functions.

Example of using System Actions to Control Users' Access to System Functions

The following example presents a very simple scenario that illustrates how system actions can be used to control users' access to system functions.

The illustration below shows the organization model of the easyAs Insurance company's Claims Management department. The department contains two sub-departments, Customer Service and Claims Handling. Each has a managerial position (Call Supervisor and Claims Manager respectively) and a staff position (Call Handler and Claims Handler). The department is headed overall by the VP for Claims Management.

The easyAs Insurance Claims Management Department



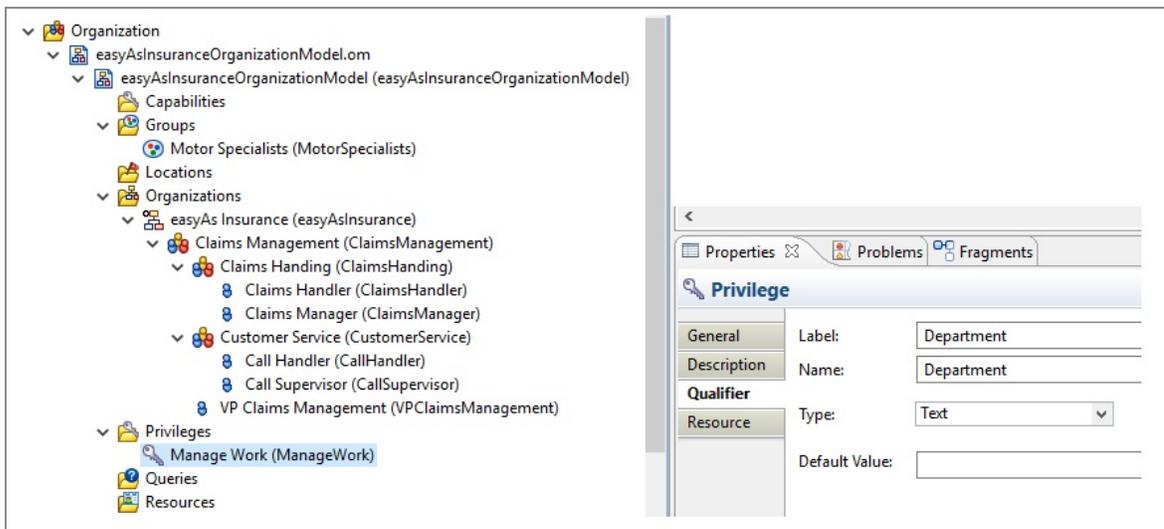
To assist in processing work efficiently, the department has the following requirements:

- The VP Claims Management should be able to view the work list of anybody in the department.

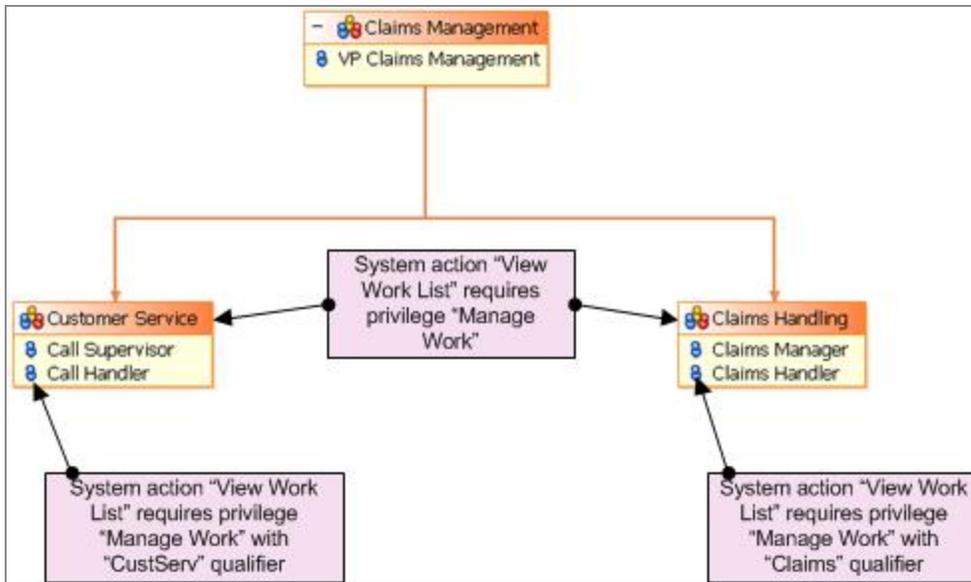
- The Call Supervisor should be able to view the work list of all Call Handlers.
- The Claims Manager should be able to view the work list of all Claims Handlers.
- The Claims Manager and Call Supervisor should not be able to view the work list of anybody in the other department.

The system action, `View Work List`, can be used to implement these requirements. By default, no user is allowed to perform this action, so its use must be authorized at the appropriate levels. This is achieved in the following way:

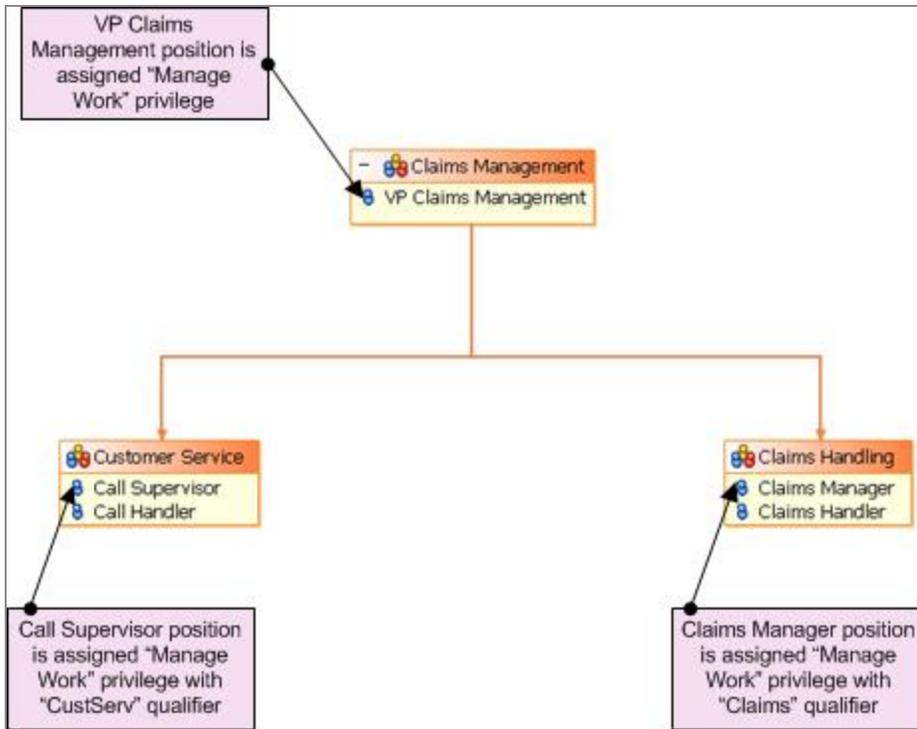
- Using the TIBCO Business Studio - BPM Edition Organization Modeler, the business analyst defines a privilege called `Manage Work`. This will be used to control access to other users' work lists. He also defines a `Department` qualifier for this privilege, the value of which will be used to identify which department the privilege applies to.



- The analyst assigns the `Manage Work` privilege to the `View Work List` system action for different entities in the organization model, as shown below.



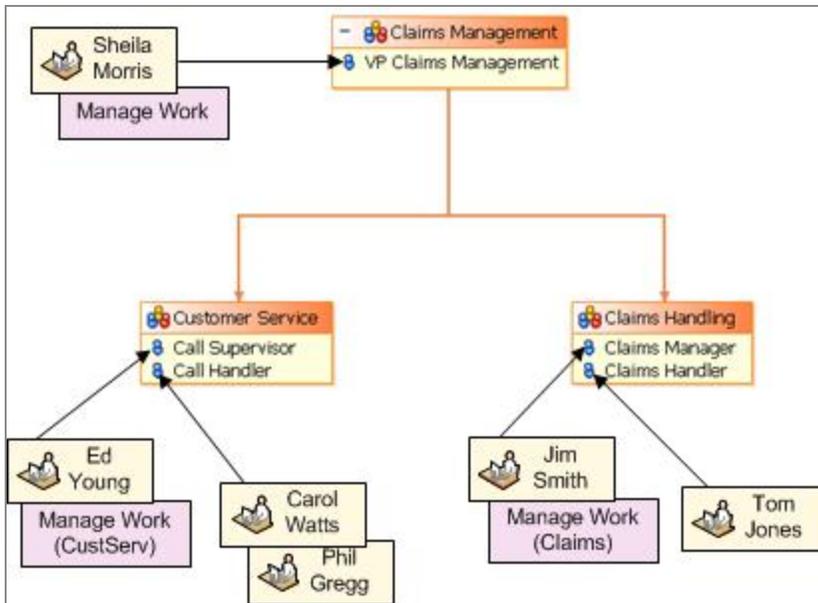
- This defines the privilege that will be required to view the work lists of users in the department, as follows:
 - To view a work list of a user in the Customer Service or Claims Handling organization unit, a user will need to hold the Manage Work privilege.
 - To view a work list of a user who holds the Call Handler position, a user will need to hold either an unqualified Manage Work privilege or hold the privilege with the Department qualifier set to CustServ.
 - To view a work list of a user who holds the Claims Handler position, a user will need to hold either an unqualified Manage Work privilege or hold the privilege with the Department qualifier set to Claims.
- The analyst assigns the Manage Work privilege to the organization model entities that need it, as shown below.



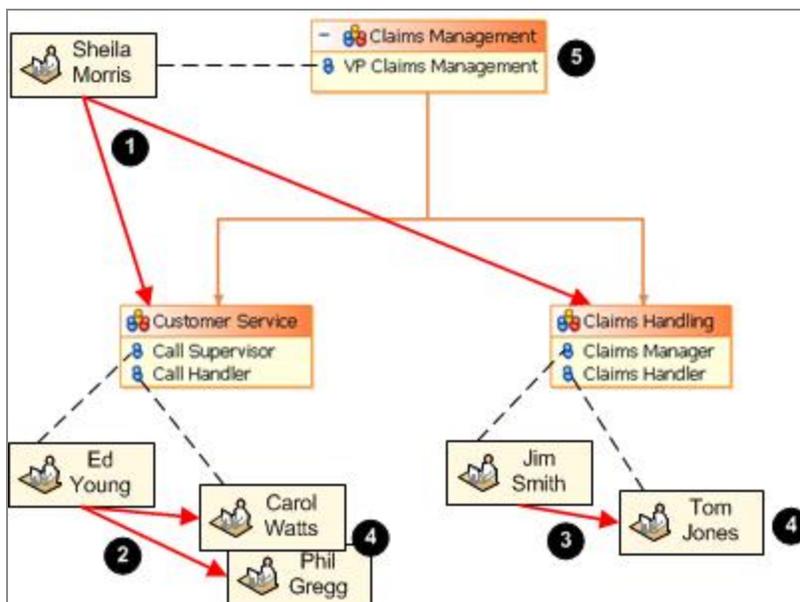
- This defines the privileges that will be inherited by users who are assigned to the following positions:
 - The user who is the VP for Claims Management will have the (unqualified) Manage Work privilege.
 - Call Supervisor users will have the Manage Work privilege, qualified with the value CustServ.
 - Claims Manager users will have the Manage Work privilege, qualified with the value Claims.

The organization model now contains the necessary information and is deployed to the TIBCO BPM Enterprise runtime.

- Using the Organization Browser, the administrator adds users from the company's LDAP directory to the appropriate positions in the organization model. These users inherit the privileges defined earlier, as shown below.



The following diagram shows how the system action and privilege settings interact at runtime, to determine which users have access to which worklists.



(1) Sheila Morris, the VP for Claims Management, can view the work lists of everybody in the Customer Services and Claims Handling departments (Ed Young, Carol Watts, Phil Gregg, Jim Smith, and Tom Jones).

(2) In the Customer Services department, Ed Young, the Call Supervisor, can view the work lists of his Call Handler reports, Carol Watts and Phil Gregg. He cannot see the work lists of anybody in the Claims Handling department.

(3) In the Claims department, Jim Smith, the Claims Manager, can view the work list of his Claims Handler report, Tom Jones. He cannot see the work lists of anybody in the Customer Service department.

(4) The Call Handlers (Carol Watts and Phil Gregg) and Claims Handler (Tom Jones) cannot view anybody else's worklists, even in their departments, as they have not been granted the Manage Work privilege.

(5) Nobody in the Customer Service or Claims Handling departments can view Sheila Morris' work list. This is because no privilege has been assigned to the `View Work List` system action for either the Claims Management organization unit or the VP Claims Management position.

System Actions and Organization Model Versions

When testing whether a user has the authorization to perform a system action, that is that the user holds the required privileges, all major versions of the organization model are taken into account.

The privileges required to perform a system action are applied on a per-major-version basis. That is, the same system action may require a different set of privileges in different major versions of the organization model, and each set of required privileges is tested independently. Similarly, a position to which a user is mapped may be granted different privileges in different versions of the organization model.

To use a system action, a user must be mapped to a position that has been granted *all* of the privileges that are required in *any* major version of the organization model.

To test for this, TIBCO BPM Enterprise examines each major version of the organization model in turn. For each major version, TIBCO BPM Enterprise gathers the required privileges defined in that version for the system action. Then:

- If no required privileges have been defined in a given major version, that version is ignored.
- If required privileges are found in a version, and the user *does not* hold all those privileges, it proceeds to test other major versions.
- If any required privileges are found in a version, and the user holds all those privileges in that version, access to the system action is **granted** and the search

stops: no further major versions of the organization model are checked.

When all of the major versions of the organization model have been checked:

- If a required privilege is defined in any major version, but the user does not qualify for access (see third bullet above), then access to the system action is **denied**.
- If there are no required privileges for the system action in *any* major version, access is granted or denied using the default access for that system action. Some system actions are open to all users by default unless any required privileges have been defined to override this default, while other system actions are denied by default.

Different Organization Models with the Same Major Version

All organization models of the same major version - for instance, versions 2.0, 2.1, 2.2, 2.2.1, and 2.3—are merged, and any required privileges set against any system action in any such version are similarly merged. Therefore, to use a system action, a user must hold all the required privileges that are defined in all organization models of the same major version.

Example of using System Actions to Control Users' Access to System Functions, continued

See: [Example of using System Actions to Control Users' Access to System Functions](#).

In the organization described in the example, changes in the business lead to the introduction of a new version of the organization model, Version 2.0, and the system action, `View Work List`, no longer requires the `Manage Work` privilege.

Carol Watts tries to view her colleague Phil Gregg's worklist. In the current version of the organization model, there are no required privileges to prevent her from doing this.

Therefore:

- TIBCO BPM Enterprise examines each major version of the organization model in turn. It starts with the current Version 2.0. No required privileges have been defined in this major version, so that version is ignored.
- Testing Version 1.0. however, TIBCO BPM Enterprise finds that a required privilege has been defined, the `Manage Work` privilege. In that same version, Carol Watts *does not* hold this privilege.
- TIBCO BPM Enterprise, therefore, does not grant Carol access but proceeds to look for other major versions to test. Finding none, it refuses Carol access to the `View`

Work List system action, even though there is no restriction in the *latest* version of the organization model to prevent her.

List of Messages

Messages generated by TIBCO BPM Enterprise events can be audited. Some of the message attributes can have more than one meaning depending on the message category.

Auditable Messages

This section lists the auditable **Audit**, **Fatal**, and **Error** messages that are generated by TIBCO BPM Enterprise.

For a list of Audit, Warn, and Error messages that may be audited and published, see the Samples and API tab on <https://docs.tibco.com/products/tibco-bpm-enterprise>. You can also download the list of auditable messages using [this link](#) directly.

Message Categories and Attribute Contents

Several attributes are available for each message category.

Standard Attributes

The attributes listed are *in addition* to the following standard attributes that are present for all messages:

-
- | | |
|----------------|-------------------|
| • creationTime | • parentContextId |
| • componentId | • messageCategory |
| • messageId | • principalId |
| • message | • principalName |
| • severity | • nodeName |
| • priority | • hostName |
-

- correlationId
 - contextId
 - hostAddress
-

Attributes and Meanings

Note that the meaning of some attributes is affected by the message category.

For example:

- If **messageCategory**='PROCESS_INSTANCE', then **managedObjectName** is the name of a process template.
- If **messageCategory**='ORGANIZATIONAL_ENTITY', then **managedObjectName** is the name of an organization model entity.

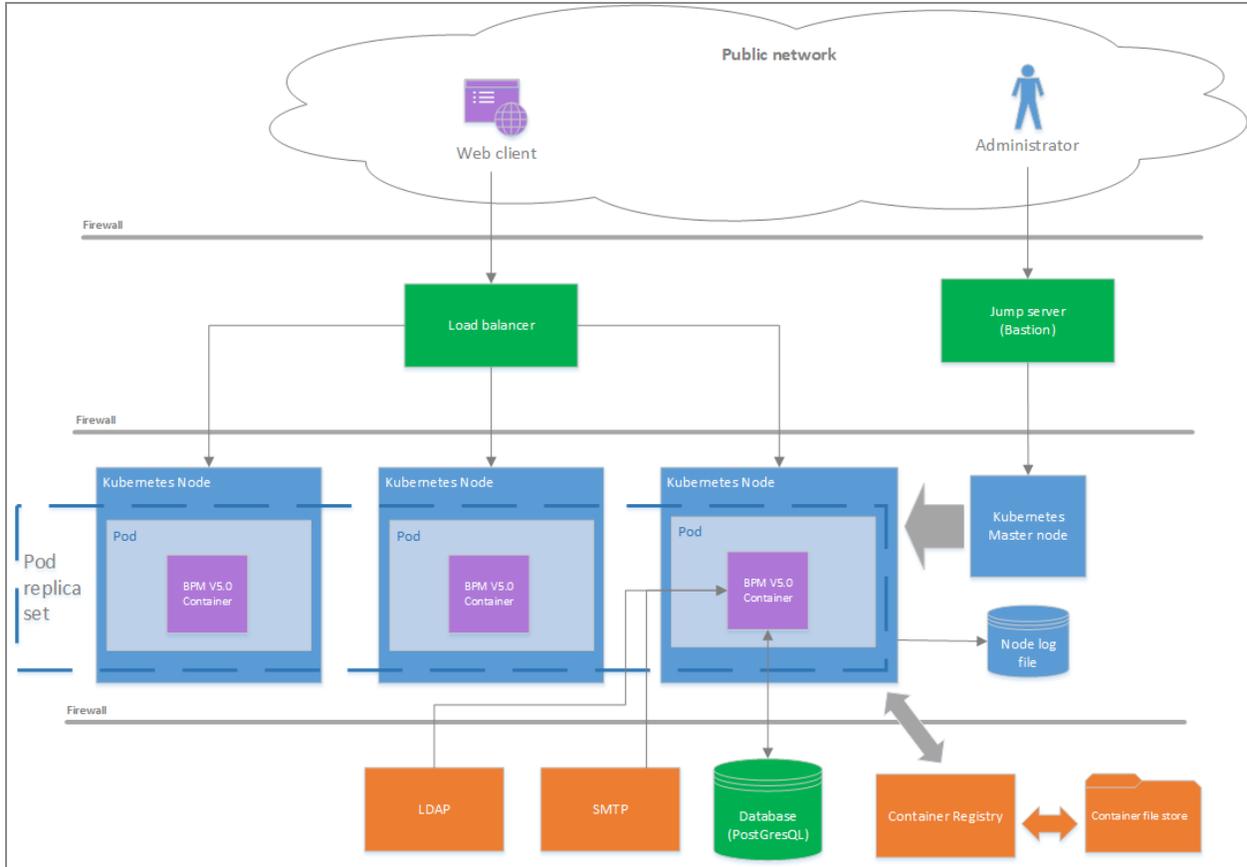
 **Note:** By default, not all attributes are audited. Messages given *in italics* in the tables are *not* audited by default.

Container Management

The Docker containers that contain TIBCO BPM Enterprise are managed by a Container Orchestration System - Kubernetes.

Kubernetes is used to manage TIBCO BPM Enterprise containers in the following ways:

- Add and remove pods from Kubernetes nodes. Replica Sets of pods across nodes can be created to scale TIBCO BPM Enterprise.
- Deploy Docker containers, containing TIBCO BPM Enterprise, to pods. As TIBCO BPM Enterprise only has a single container type, each pod will contain only a single container and the associated resources (database shared resources, IP address, and so on) required by TIBCO BPM Enterprise.
- Startup and shutdown of Docker containers containing TIBCO BPM Enterprise.
- Basic health monitoring. This is accomplished using Kubernetes cAdvisor, which monitors container metrics, as well as application metrics.
- Configuration injection. TIBCO BPM Enterprise configuration is injected into the container on startup. Some of this configuration is written to the database to ensure consistency across all containers.



The following illustrates how TIBCO BPM Enterprise is deployed in Kubernetes.

Pods are created and deployed in Kubernetes nodes. Each pod contains a single TIBCO BPM Enterprise container. Additional pods can be added to a node to scale the system. The system can also be scaled by creating a replica set across multiple Kubernetes nodes.

There is a single TIBCO BPM Enterprise service per node, which exposes the product REST API.

For additional information, see *TIBCO BPM Enterprise Administration*.

Database Support

All Container Orchestration System nodes, and TIBCO BPM Enterprise containers, use the same database.

For the list of supported databases, see System Requirements (Kubernetes Profile) of the *TIBCO® BPM Enterprise Installation*. Scripts are provided to set up the database during installation.

Connection details for the database are passed into TIBCO BPM Enterprise upon startup using injected configuration.

There is no support for XA transactions. All global data (case data) is stored in the TIBCO BPM Enterprise database.

Statistical and event data can be accessed from the database via reporting. If you manage your own database you can directly access the database tables. For more information, see [Event Collector](#). In addition to this, the JDBC task activity must be idempotent.

Appendix: Utility Commands

To use the Utility Commands, see the following table.

Usage

```
docker run -it --rm tibco/bpm/utility:5.6.0 utility -options
```

Options:

Options	Arguments	Description
-setLogging	logRoot=<TRACE DEBUG INFO WARN ERROR> logLevel=<Log Level String> logFormat=<COLUMNS RFC5424 JSON> [--verbose]	Sets the system log configuration.
-showLogging		Shows the currently configured log level for the system.
-traceEvents	eventTypes=<GENERIC QUEUE ALL> waitTime=<Num Seconds>	Traces the system event notifications.
-dbConfig	url=<Database URL> username=<DB Username> password=<DB Password>	Connection information required for BPME database.
-setupDatabase	[execute] [--verbose]	Produces the database schema create upgrade script and optionally runs it against the configured database. --verbose forces output if SQL commands are in 'execute' mode.
-showAdminUser		Selects and displays current details of

Options	Arguments	Description
		the admin user.
-setupAdminUser	ldapAlias=<Alias> ldapDn=<LDAP DN> displayName=<Display Name> [--verbose]	Configures the tibco-admin user to a custom LDAP or display name. --verbose selects and displays current details after the update.
-decode	<uri>	Decode a PVM URI, returning the database Id, object type, and version found. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-encode	<objectType> <dbId> [ownerId]	Encode a PVM URI using the database Id for the given object type and version. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-dumpInfo		Dump the database information. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-dumpHalted		Dump information on halted instances. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-dumpProcess	<instanceId uri>	Dump information for the given instance, and any related instances. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.

Options	Arguments	Description
-diagnostics		Run diagnostics against the BPM system. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-purgeInstance	<instanceId URI> [--full] [--dryRun] [--ec] [--sc] [--verbose]	Purge instances for a given instance Id/URI. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-purgeProcess	<processName> [version=<version>] [batchSize=<batchSize>] [--dryRun] [--verbose]	Purge instances for a given process. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-sendEvent	<taskId URI> <eventCode> [--dryRun]	Send the required event to the given task. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-purgeWorkitem	<workitemId> [--dryRun] [--verbose]	Purge work item. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-alterCounter	<taskId URI> <counterValue> [--dryRun]	Increments the counter by the value for the given task. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-invokeExternal	<taskId URI> [--dryRun]	Invoke an EXTERNAL(20) event on the given task. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.
-cleanup	[--dryRun]	Perform cleanup. Use this parameter only if advised by TIBCO Support. For more details, contact TIBCO Support.

Argument values with spaces should be enclosed in single quotes, for example,
`ldapDn='UID=admin, OU=system'`.

TIBCO Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the [Product Documentation website](#), mainly in HTML and PDF formats.

The [Product Documentation website](#) is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The documentation for this product is available on the [TIBCO® BPM Enterprise Product Documentation](#) page.

How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our [product Support website](#).
- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the [product Support website](#). If you do not have a username, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature

requests from within the [TIBCO Ideas Portal](#). For a free registration, go to [TIBCO Community](#).

Legal and Third-Party Notices

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, Business Studio, TIBCO Business Studio, and Spotfire are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: <https://scripts.sil.org/OFL>

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file for the availability of a specific version of Cloud SG software on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "README" FILES.

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.cloud.com/legal>.

Copyright © 2015-2024. Cloud Software Group, Inc. All Rights Reserved.