Member Manager staff plug-in provider Version 6.0.2

The Member Manager staff plug-in is a staff plugin-in for the exclusive usage with IBM WebSphere Portal. This document provides detailed information about the Member Manager staff plug-in that must be used with Version 6.0.2 of WebSphere Process Server for Business Process Support in WebSphere Portal.

Note that the plug-in is not installed as part of a WebSphere Process Server 6.0.2 installation. It must be installed in addition to such an installation. A standard install of WebSphere Portal does install the Member Manager staff plug-in provider that is included in the portal shipment. This can be either version 6.0.2 or an earlier version of the plug-in. In the latter case, you must upgrade the plug-in to version 6.0.2 .

Table of contents

1 Business Process Choreographer staff plug-in providers overview

Business Process Choreographer allows the operation of business processes defined according to the Business Process Execution Language standard. Such processes are modeled, deployed, and run on top of the WebSphere Process Server product as well as in the business process support installed with WebSphere Portal.

A business process is a design for how a series of activities is done. Staff activities are one type of activity. They are assigned to people through work items. Staff activities can be almost any business task such as completing a form, approving a document or drawing, and writing a letter. When a process is started, work items are created for the potential owners.

When a user claims an activity, they become the owner of that activity. Only they can work on that activity in that particular instance of the process. If the work is complex or involved, the user can save intermediate stages of the work. When the work is done, the user completes the activity. The resulting information is saved and is then available to subsequent activities in the process.

Humans involved in staff activities assume one of several possible roles. Examples for such roles include the following:

- **Potential Owner:** Permitting a human to claim and complete a staff activity
- **Editor:** Allowing a human to contribute to a staff activity
- **Reader:** Allowing a human to view the data of a staff activity

During the modeling of a business process, each staff activity can be associated with one or more of these roles.

For each role, delimit a set of user IDs to indicate which users can assume what role. This delimitation is defined via so-called Staff Verbs, which are specified by the process modeler. Process Choreographer comes with a pre-defined set of Staff Verbs, which correspond to user selections such as the following:

- The user with distinguished name A using the "Users" staff verb
- The user with the short name B using the "Users by User ID" staff verb
- The users which are part of user group C using the "Group Members" staff verbs

During the execution of a staff activity, Process Choreographer has to resolve the associated Staff Verbs, i.e. it has to determine the set of users defined by the verbs. It does so by mapping the verbs to user repository queries such as the following:

- Look up user with distinguished name A
- Look up the users with of user group C

Such queries are specific to the employed user repository implying that Process Choreographer has to perform mappings between Staff Verbs and specific user repository queries.

In Process Choreographer, this functionality is contained in user repository specific modules, so-called staff resolution plug-ins. Process Choreographer comes with the following three plug-ins:

- **An LDAP staff provider plug-in:** For mapping to LDAP server queries
- **A system staff provider plug-in:** For mapping to an OS user repository
- **A user registry staff provider plug-in**

When using the IBM® WebSphere® Portal Member Manager module to access a user repository, the Member Manager staff plug-in provider must be installed.

2 Upgrading an existing version of Member Manager staff plug-in provider to the plug-in version 6.0.2

2.1 Scenario without network deployment

If you already use an earlier version of the Member Manager staff plug-in provider you must perform the following steps:

1.) Determine the node on which the Member Manager staff plug-in is configured:
   a. For a standard portal installation, this is the node hosting WebSphere Portal
   b. For non-standard installations, this is the node hosting the server on which Business Process Choreographer has been configured.

2.) Copy the file `lib/bpestaffmembermanager.jar` from the Member Manager staff plug-in V6.0.2 package to the *app_server_root*`/lib` directory on the target machine.

3.) If you want to use any new V 6.0.2 staff verbs (see the corresponding section) you must merge the file ProcessChoreographer/`MemberManagerTransformation.xsl` from the Member Manager staff plug-in V6.0.2 package with the .xsl file that is used by your staff plug-in provider configuration:
   a. Using the WAS Administrative Console, locate the .xsl file that is being used by clicking:
      Resources -> Staff-Plug-in provider -> Member Manager Staff Resolution Plug-in -> Staff plug-in configuration -> XSL Transform File
      Note: If you created a new Staff Plug-in provider, follow the corresponding path in the Admin Console, for example,
      Resources -> Staff-Plug-in provider -> *My Member Manager Staff Plug-in* -> Staff plug-in configuration -> XSL Transform File
   b. Make a copy of the V6.0.2 file `MemberManagerTransformation.xsl` that is provided in the *app_server_root*`/ProcessChoreographer/Staff/` directory and rename it, for example, to MyMMTransformation_602.xsl
   c. Apply the necessary adaptations to your copy as necessary for the user repository in use (for example LDAP). For some examples, see the section on ‘Staff verb semantics’
   d. Modify the XSL Transform file name in the plug-in configuration to reflect the path of the .xsl file copy. For example, set
      Resources -> Staff-Plug-in provider -> Member Manager Staff Resolution Plug-in -> Staff plug-in configuration -> XSL Transform File
      to reflect the new name
      ${WAS_INSTALL_ROOT}/ProcessChoreographer/Staff/
      MyMMTransformation_602.xsl

2.2 Scenario with network deployment

Apply the previous steps to install the plug-in throughout the cell. Perform the steps for the deployment manager and all nodes that host servers on which Business Process Choreographer has been configured.

3 Installing Member Manager staff plug-in provider

3.1 Scenario without network deployment

IBM® WebSphere® Portal uses Member Manager to provide user repository queries. If you are using WebSphere Portal and Process Choreographer, you must provide Staff verb resolution via Member Manager; the Member Manager staff plug-in provider resolves the queries.

As the Business Process Container makes use of the Member Manager staff plug-in, the plug-in must be installed and setup on the node that hosts the server for which the Business Process Container is configured.

If you use the standard portal installation, the process container is configured for the portal server and the Member Manager staff plug-in is created automatically on the node on which the portal server resides.

If you perform a non-standard portal installation or change the topology afterwards in a way that the process container is configured on a different server than the portal server, you must manually setup the plug-in on the node that hosts the server for which the container is configured.
Note: In this case, the different servers must be part of the same single sign-on domain and share the same user repository.

If the process container is configured on an application server that does not contain a portal profile, copy the required files from the portal installation to the node with the process container.

- Copy the file `lib/bpestaffmembermanager.jar` from the Member Manager staff plug-in V6.0.2 package to the *app_server_root*`/lib` directory on the target machine.
- Copy *portal_server_root*`/lib/wmm.jar` to the *app_server_root*`/lib` directory on the target machine.
- Copy the file `ProcessChoreographer/MemberManagerTransformation.xsl` from the Member Manager staff plug-in V6.0.2 package to the *app_server_root*`/ProcessChoreographer/Staff/` directory on the target machine.
- Copy the file `ProcessChoreographer/bpestaffmembermanager.jacl` from the Member Manager staff plug-in V6.0.2 package to the *app_server_root*`/ProcessChoreographer` directory on the target machine.
- Copy the file `ProcessChoreographer/remove_bpestaffmembermanager.jacl` from the Member Manager staff plug-in V6.0.2 package to the

app_server_root/ProcessChoreographer directory on the target machine.

To configure the Member Manager staff plug-in provider, perform the following:

1. Open a command prompt on the node hosting the process container and change to the /bin directory of the according application server profile.
2. Run the following command.
3.
   ```
   wsadmin -f
   app_server_root/ProcessChoreographer/bpestaffmembermanager.jacl
   -conntype SOAP -user admin_ID -password admin_password ejbHome
   admin_ID admin_password
   ```

   where

   - *ejbHome* is the name of the Member Manager EJB home
   - *admin_ID* (both occurrences) is the user ID for the application server administrator
   - *admin_password* (both occurrences) is the password for the application server administrator
   -

   To achieve communication between servers that are not in a managed cell, define the ejbHome using one of the following options.

   - `corbaname:iiop:host:port#ejb/MemberServiceHome`
   - Create an indirect binding using the application server administrative console and then use the value that you defined for the Name in Name Space property, for example: `my/MemberServiceHome`.
4. Follow these steps to enable identity assertion:
   a. Use your browser to log in to the application server administrative console. For example: http://www.example.com:9060/ibm/console.
   b. Select Security > Global Security > Authentication Protocol > CSIv2 Inbound authentication.
   c. Select Identity assertion.
   d. Click OK.
   e. Select Security > Global Security > Authentication Protocol > CSIv2 Outbound authentication.
   f. Select Identity assertion.
   g. Click OK and Save.
   h. Restart the application server.

i. Run the wsadmin scripting client with these options.

```
wsadmin -f
app_server_root/ProcessChoreographer/util/refreshStaffQuery
.jacl -server servername
```

For more information about this command, see Using scripting (wsadmin) in the information center for IBM WebSphere Application Server.

Follow these steps to verify that the Member Manager staff plug-in provider is installed:

1. Open the **WebSphere Application Server administration console**; for example, `http://hostname:9060/ibm/console`.
2. Log in as administrator.
3. Click Resources > Staff plug-in Provider.

   Note: You should see four installed staff plug-in providers; one is the Member Manager Staff Resolution plug-in Provider and the others are default Process Choreographer plug-ins.

4. Restart the node so that the changes take effect

3.2 Scenario with network deployment

Apply the following steps to install the plug-in throughout the cell.

1. Perform these steps for the deployment manager and all nodes on machines that do not contain a portal profile. Copy these required files from the portal installation to the according deployment manager or process container node.
   - Copy the file `lib/bpestaffmembermanager.jar` from the Member Manager staff plug-in V6.0.2 package to the `app_server_root/lib` directory on the target machine.
   - Copy `portal_server_root/lib/wmm.jar` to the `app_server_root/lib` directory on the target machine.
   - Copy the file `ProcessChoreographer/MemberManagerTransformation.xsl` from the Member Manager staff plug-in V6.0.2 package to the `app_server_root/ProcessChoreographer/Staff` directory on the target machine.
2. From a command prompt, change to the `portal_server_root/config` directory and run the following command .

   ```
   WPSconfig.{bat|sh} action-create-new-bpe-wmmplugin-config  -
   DWmmEjbName=wmmejbname -DNodeName=nodename
   ```

   where

- *wmmejbname* is the name of the Member Manager EJB home. In a managed cell, this could be defined as

  ```
  cell/nodes/portal_nodename/servers/portal_servername/ejb/MemberServiceHome
  ```

  to address one specific server or

  ```
  cell/clusters/portal_clustername/ejb/MemberServiceHome
  ```

  to address a portal cluster.

- *nodename* is the name of the node to which the staff plug-in should be installed.

Note: The administrative user ID and password must be set in wpconfig.properties before running this command.

3. Follow these steps to enable identity assertion:
   a. Use your browser to log in to the application server administrative console. For example: http://www.example.com:9060/ibm/console.
   b. Select Security > Global Security > Authentication Protocol > CSIv2 Inbound authentication.
   c. Select Identity assertion.
   d. Click OK.
   e. Select Security > Global Security > Authentication Protocol > CSIv2 Outbound authentication.
   f. Select Identity assertion.
   g. Click OK and Save.
   h. Restart the application server.
   i. Run the wsadmin scripting client with these options.

   ```
   wsadmin -f app_server_root/ProcessChoreographer/util/refreshStaffQuery.jacl -server servername
   ```

   For more information about this command, see Using scripting (wsadmin) in the information center for WebSphere Application Server.

1. Follow these steps to verify that the Member Manager staff plug-in provider is installed:
   a. Open the **WebSphere Application Server administration console** on the deployment manager; for example, `http://hostname:9060/ibm/console`.
   b. Log in as administrator.
   c. Click Resources > Staff plug-in Provider.
   d. Set the Scope to one of the federated nodes.

   e. Ensure that the Member Manager staff plug-in provider is registered for the node.

2. Restart the deployment manager and all modified nodes.

4 Configuring server communication end points

If you experience problems when restarting `WebSphere_Portal` without restarting the server that hosts the Business Process Container (and the plug-in), you should associate the following end points of `WebSphere_Portal` with static port values

1. In case CSIv2 is used as security protocol between the servers, set static port values for the end points:
2. `CSIv2_SSL_MUTUALAUTH_LISTENER_ADDRESS`
   `CSIv2_SSL_SERVERAUTH_LISTENER_ADDRESS`

3. In case SAS is used as security protocol between the servers, set a static port value for the end point:

   `SAS_SSL_SERVERAUTH_LISTENER_ADDRESS`

For background information on these protocols and their settings, please refer to the WebSphere Process Server Version 6.0 Information Center at http://www.ibm.com/software/integration/wps/library/.

For setting static port values for server end points, proceed as follows:

1. In the administrative console, click Servers > Application Servers > WebSphere_Portal.
2. Under Business Integration > Additional Properties, select End Points.
3. For every required end point (i.e. `CSIv2_SSL_MUTUALAUTH_LISTENER_ADDRESS`)
   a. Select the end point.
   b. Configure a static port value (i.e. 8355).
   c. Click **Apply**.
4. Click **OK**.
5. Click **Save** in the messages window of the Administrative Console to save the modified settings; be sure to confirm the save.
6. Restart both servers (WebSphere_Portal and server1).

5 Default and new Member Manager staff plug-in provider configurations

The Member Manager staff plug-in provider comes with a default configuration that is ready to use. You can leave the configuration as is or you can make changes via the WebSphere Application Server Admin Console. You can also create multiple configurations for each plug-in, which are distinguished by their JNDI name.

Changing the default configuration settings

Follow these steps to change the default configuration settings:

1. Open the WebSphere Application Server administration console. For example, `http://`*`hostname`*`:9060/admin`.
2. Log in as administrator.
3. Click **Resources>Staff plug-in Provider**.
4. Click **Member Manager Staff Resolution plug-in**.
5. Select **Staff plug-in Configuration**.
6. Select **WMM Staff plug-in Configuration for WPS**.
7. Click **Custom Properties**.
8. To change a setting, click the property name, enter a value, and click **OK**. See the following table for properties:

plug-in properties	Required/Optional	Comments
AuthenticationAlias	Required	Used to access the Member Manager service. The alias name is set here and can be found at **Security>JAAS Configuration>J2C Authentication Data**. By default, the value points to the pre-configured Member Manager access alias.
ContextFactory	Optional	Sets the Java Naming and Directory Interface (JNDI) context factory; for example, com.ibm.websphere.naming.WsnInitialContextFactory
DefaultSearchTimeout	Optional	Controls when to terminate a search operation.
WMMEJBHomeName	Optional	The JNDI EJB home interface where the WMM EJB is found.

9. Click **Save** to apply the changes.
10. Stop and then restart the server to activate the plug-in.

    Note: See Troubleshooting for additional information.

Creating new configurations

For any plug-in, you can create more than one configuration. Configurations are distinguished by their JNDI name and a business process has to be associated with a plug-in configuration name at process modeling time. By selecting a specific configuration name, varying properties of the same plug-in can be enforced. Follow these steps to create new configurations:

1. Create a copy of the file `app_server_root/ProcessChoreographer/Staff/MemberManagerTransformation.xsl`.
2. Edit the copy, and modify the search attributes in the "Global variables" section for your user registry before deploying your process.
3. Open the WebSphere Application Server administrative console by accessing the following URL in a browser: `http://hostname.example.com:9060/admin`.
4. Log in as **administrator**.
5. Click **Resources>Staff plug-in Provider**.
6. Click **Member Manager Staff Resolution plug-in**.
7. Select **Staff plug-in Configuration**.
8. Click **New**.
9. Click **Browse**, and select the copy of the `MemberManagerTransformation.xsl` file that you created previously.
10. Click **Next**.
11. Enter an **administrative name** for the staff plug-in provider.
12. Enter a **description**.
13. Enter the Java Naming and Directory Interface (JNDI) name for the business process to use when referencing this plug-in; for example, `bpe/staff/wpswmmconfiguration2`.
14. Click **Apply**.
15. Click **Custom Properties**.
16. See Default and new Member Manager staff plug-in provider configurations to change the default settings of your new configuration.

6 New staff verbs in V6.0.2

The following additional staff verbs are included in V6.0.2:

- Group
- Group Members without Named Users
- Group Members without Filtered Users
- Users by User ID without Named Users

7 Staff verb semantics

To a large extent the semantics of the staff verbs is independent of the plug-in to be selected. Nevertheless, some specific semantics exist which are pointed out below for the Member Manager.

The following set of verbs is supported by staff plug-ins:

- Department members
- Everybody
- Group
- Group members
- Group Members without Named Users
- Group Members without Filtered Users
- Group search
- Manager of employee
- Manager of employee by user ID
- Native query
- Nobody
- Person search
- Role members
- Users
- Users by user ID
- Users by User ID without Named Users

Note: Do not use the original MemberManagerTransformation.xsl file when applying the changes described in the examples below. Use a copy of this file instead and reference the copy from the plug-in configuration.

Department members

Use this verb to define a query to retrieve the members of a department.

The verb translates into a search for all entries of type (i.e. LDAP objectclass) "person", which are located under a predefined search base, for example `"o=deptName,cn=departments,dc=mycomp"`, and retrieving a predefined attribute, for example `"uid"`.

A process modeler can set a number of parameters when selecting the verb for an activity role. Except for the department names, none of the parameters are evaluated during the staff verb resolution.

Parameter	Use	Type	Supported by Member Manager	Description
DepartmentName	Mandatory	String	Yes	Department name of the users to retrieve
IncludeNestedDepartments	Mandatory	Boolean	No	Not evaluated
Domain	Optional	String	No	Not evaluated
`AlternativeDepartmentName1`	Optional	String	LDAP	Second department to include
AlternativeDepartmentName2	Optional	String	LDAP	Third department to include

The specification of the search parameters (type, search base, retrieved attribute) is fixed in the transformation file used by the plug-in configuration, for example `MemberManagerTransformation.xsl` located in the *app_server_root*`/ProcessChoreographer/Staff` directory. If changes are needed, the `.xsl` file has to be adapted accordingly.

Example (change user return attribute to "cn", user search attribute to "userid", suffix for department names to "`dc=areas,dc=mycomp`"):

```
        <xsl:template name="DepartmentMembers">
        ...
        <swmm:search>
                       <xsl:attribute name="searchBase">
                              o=<xsl:value-of
select="$deptname"/>,cn=areas,dc=mycomp
                       </xsl:attribute>
                       <xsl:attribute
name="returnType">person</xsl:attribute>
                       <xsl:attribute
name="returnAttribute">cn</xsl:attribute>
                       <xsl:attribute
name="searchAttribute">userid</xsl:attribute>
                       <xsl:attribute
name="operator">notNULL</xsl:attribute>
```

```
                        <xsl:attribute
name="referenceValue">dummy</xsl:attribute>
                        <xsl:attribute
name="referenceType">String</xsl:attribute>
        </swmm:search>
        ...
        </xsl:template>
```

Everybody

Use this verb to assign a work item to every user authenticated by the WebSphere Application Server. This verb has no parameters; it is supported by all plug-ins including the Member Manager plug-in.

Group

Use this verb to define a query to authorize the members of a group. This verb is similar to the “Group Members” verb, in that it amounts to authorizing all members of a group for a certain role. However, the run-time behaviour is different:
“Group Members” computationally results in a list of user Ids, against which, a given userID is checked at run-time.
“Group” does not result in a list of user IDs stored at run-time. For a logged in user, the user ID is used to derive the names of the groups it belongs to. The set of these group names is checked against the group denoted by a “Group” staff verb.
The “Group” staff verb has been introduced to avoid heavy computational and storage loads for large user groups.

Parameter	**Use**	**Type**	**Description**
GroupId	Mandatory	string	The name of the group of users to authorize.

Group Members

Use this verb to define a query to retrieve the members of a group. For the Member Manager plug-in, a group is associated with the semantics defined by WebSphere Portal for its user groups. According to this, a user group can consist of users and subgroups that also consist of users and other subgroups. The search for the users of a group can be defined to be non-recursive or recursive. In the former case, only the users are retrieved directly belonging to the specified group. In the latter case, the users of all subgroups are included as well.

A process modeler can set a number of parameters when selecting the verb for an activity role:

Parameter	Use	Type	Supported by Member Manager	Description
GroupName	Mandatory	String	Yes	Group name of the users to retrieve
IncludeSubgroups	Mandatory	Boolean	No	Specifies whether nested subgroups are considered in the query
Domain	Optional	String	No	Not evaluated
AlternativeDepartmentName1	Optional	String	LDAP	Second department to include
AlternativeDepartmentName2	Optional	String	LDAP	Third department to include

Example (change user return attribute to "cn", suffix for group names to "`dc=areas,dc=mycomp`"):

```
        <swmm: usersOfGroup>
                <xsl:attribute name="id">
                        o=<xsl:value-of
select="$groupname"/>,dc=areas,dc=mycomp<xsl:value
                </xsl:attribute>
                <xsl:attribute name="idType">memberDN</xsl:attribute>
                ...
                <xsl:attribute name="attribute">cn</xsl:attribute>
        </swmm:usersOfGroup>
```

Group Members without Named Users

Use this verb to define a query to retrieve all the members of a group except for the explicitly named users.

Parameter	Use	Type	Description
GroupName	Mandatory	string	Group name of the users to retrieve. Supports custom properties that are evaluated at run time.
IncludeSubgroups	Mandatory	boolean	Specifies whether nested subgroups are considered in the query.
NamedUsers	Mandatory	string	The user ID of the users to exclude from the retrieved group members

Parameter	Use	Type	Description
			list. Supports context variables and custom properties, such as `%htm:task.originator%`

Group Members without Filtered Users

Use this verb to define a query to retrieve all members of a group except for a set of users that is defined by an LDAP search filter.

Parameter	Use	Type	Description
GroupName	Mandatory	string	Group name of the users to retrieve.
IncludeSubgroups	Mandatory	boolean	Specifies whether nested subgroups are considered in the query.
FilterAttribute	Mandatory	string	Name of the attribute to use in the filter.
FilterValue	Mandatory	string	Filter value to use in the filter.

Group Search

Use this verb to search for a group based on an attribute match and to retrieve the members of the group.

This verb implies two hidden queries. First, all user groups are retrieved featuring a specified attribute and a specified value, for example Business Type equaling Finance. Then all users belonging to these groups are retrieved.

A process modeler can set a number of parameters when selecting the verb for an activity role:

Parameter	Use	Type	Supported by Member Manager	Description
GroupID	Optional	String	Yes	The group ID of the users to retrieve
Type	Optional	String	Yes	The group type of the users to retrieve
IndustryType	Optional	String	Yes	The industry type of the group to which the users belong
BusinessType	Optional	String	Yes	The business type of the group to which the users belong

Parameter	Use	Type	Supported by Member Manager	Description
GeographicLocation	Optional	String	Yes	An indication of where the users are located
Affiliates	Optional	String	Yes	The affiliates of the users
DisplayName	Optional	String	Yes	The display name of the group
Secretary	Optional	String	Yes	The secretary of the users
Assistant	Optional	String	Yes	The assistant of the users
Manager	Optional	String	Yes	The manager of the users
BusinessCategory	Optional	String	Yes	The business category of the group to which the users belong
ParentCompany	Optional	String	Yes	The parent company of the users

One attribute (Type, Industry Type, ...) should be set at a time. Only the first attribute in the list will be evaluated.

The specification of the attribute to be retrieved for the group members (i.e. uid) is fixed in the transformation file used by the plug-in configuration, for example `MemberManagerTransformation.xsl`, located in the directory *app_server_root*`/ProcessChoreographer/Staff`. If changes are needed, the `.xsl` file has to be adapted accordingly.

Example (change group return attribute to "groupid", group name suffix to "`dc=areas,dc=mycomp`", user return attribute to "userid"):

```
        <xsl:template name="GroupSearch">
        ...
        <swmm:search>
                <xsl:attribute name="returnType">group</xsl:attribute>
                <xsl:attribute
name="returnAttribute">groupid</xsl:attribute>
                <xsl:attribute
name="searchAttribute">"$searchparam"/></xsl:attribute>
                <xsl:attribute name="operator">equal</xsl:attribute>
                <xsl:attribute name="referenceValue"><xsl:value-of
select="staff:parameter"/></xsl:attribute>
        </swmm:search>
        ...
        <swmm:usersOfGroup>
                <xsl:attribute
name="id">cn=%groupname%,dc=areas,dc=mycomp</xsl:attribute>
                <xsl:attribute name="idType>memberDN</xsl:attribute>
```

```
                        <xsl:attribute
name="recursive">yes</xsl:attribute>
                        <xsl:attribute
name="attribute">userid</xsl:attribute>
        <swmm:usersOfGroup>
        ...
        </xsl:template>
```

Manager of Employee

Use this verb to retrieve the manager of a person using the person's name. A process modeler can set a number of parameters when selecting the verb for an activity role:

Parameter	Use	Type	Supported by Member Manager	Description
EmployeeName	Mandatory	String	Yes	The name of the employee whose manager is retrieved
Domain	Optional	String	No	Not evaluated

The specification of the attribute to be retrieved as manager attribute (i.e. "manager") as well as the attribute to be retrieved for the manager entry (i.e. "uid") are fixed in the transformation file used by the plug-in configuration, for example `MemberManagerTransformation.xsl` located in the *app_server_root*`/ProcessChoreographer/Staff` directory. If changes are needed, the `.xsl` file has to be adapted accordingly.

Example (change attribute storing the manager DN to "firstlinemanager", user return attribute to "userid"):

```
        <xsl:template name="ManagerofEmployee">
        ...
        <swmm:user>
                <xsl:attribute name="id">
                        <xsl:value-of
select="staff:parameter[@id='EmployeeName']"/>
                </xsl:attribute>
                <xsl:attribute name="idType">memberDN</xsl:attribute>
                <xsl:attribute
name="attribute">firstlinemanager</xsl:attribute>
        </swmm:user>
        ...
        <swmm:user>
                <xsl:attribute name="id">%manager%</xsl:attribute>
                <xsl:attribute name="idType">memberDN</xsl:attribute>
                <xsl:attribute name="attribute">userid</xsl:attribute>
        </swmm:user>
        ...
        </xsl:template>
```

Manager of Employee by User ID

Use this verb to retrieve the manager of a person using the person's user ID.

This verb is similar to Manager of Employee except that it takes a user ID as input instead of a full name, for example `wpsadmin` instead of `"uid=wpsadmin,cn=users,dc=mycomp"`, and permits the use of context variables.

Parameter	Use	Type	Supported by Member Manager	Description
EmployeeUserID	Mandatory	String	Yes	The user ID of the employee whose manager is retrieved. Supports context variables, such as `%wf:process.starter%`
Domain	Optional	String	No	Not evaluated

Native Query

Use this verb to define a search query based on specific parameters.

The result of the query is a set of users represented by a specified user attribute, which is to be returned, for example "cn".

The query allows the following three search constraints:

- A **search base** indicating the search space within which the search is to be conducted
- A **search type** indicating the type of objects to search on
- A **search condition** indicating a query value which needs to match the content of a specified search attribute

The following parameters permit the specification of the constraints:

Parameter	Use	Type	Supported by Member Manager	Description
QueryTemplate	Mandatory	String	Yes	The query template to use for the query. Has to be one of the following: • search • user • usersOfGroup
Query	Mandatory	String	Yes	The value to find for the search attribute

Parameter	Use	Type	Supported by Member Manager	Description
AdditionalParameter1	Mandatory	String	Yes	The operator to use when evaluating the search attribute contents against the queried value. Has to be one of the following: • equal • greaterThan • greaterOrEqual • lessOrEqual • like • lessThan • notEqual • notNull
AdditionalParameter2	Mandatory	String	Yes	Search attribute to use for finding entries, for example "uid"
AdditionalParameter3	Mandatory	String	Yes	The name of the attribute to retrieve for found users, for example "cn"
AdditionalParameter4	Mandatory	String	Yes	Specifies the type of entries to search on. Has to be one of the following: • person • group • groupRecursive Note: Used in template "search" only.
AdditionalParameter6	Optional	String	Yes	The search base to use, for example "cn=users,dc=mycom" Note: Used in template "search" only.

Example 1:

```
Query:                                 peter
Query template:                search
```

```
        AdditionalParamter1:   equal
        AdditionalParameter2:  uid
        AdditionalParameter3:  uid
        AdditionalParameter4:  person
        AdditionalParameter5:  cn=users,dc=mycom

        has to be read as:
                search (from Query template)
                                on entries of type person (from
AdditionalParameter4)
                                under search base cn=users,dc=mycom (from
AdditionalParameter5)
                                for entries matching search condition
                                                uid (from
AdditionalParameter2)
                                                equal (from
AdditionalParamter1)
                                                peter (from Query)
                and retrieve for all found (user) entries
                                the value of attribute uid (from
AdditionalParameter3)
```

Example 2:

```
        Query:                                  department1
        Query template:                 search
        AdditionalParameter1:  equal
        AdditionalParameter2:  cn
        AdditionalParameter3:  uid
        AdditionalParameter4:  group
        AdditionalParameter5:  cn=groups,dc=mycom

        has to be read as:
                        search (from Query template)
                                        on entries of type group (from
AdditionalParameter4)
                                        under search base
cn=group,dc=mycom (from AdditionalParameter5)
                                        for entries matching the search
condition
                                                        cn (from
AdditionalParameter2)
                                                        equal (from
AdditionalParamter1)
                                                        department1 (from
Query)
                        and retrieved for all users contained in the
found group(s)
                                        the value of attribute uid (from
AdditionalParameter3)
```

Note: When searching on objects of type "groupRecursive", the users of found group(s) will be considered for retrieval as well as the users that are part of subgroups of the group(s).

Query templates

The following three templates can be used:

- The **search** template is the most general one, allowing use of all mentioned parameters. All parameters except `AdditionalParameter5` (search base) have to be specified. If no search base is specified, the search base defined by Member Manager for users (`cn=users,dc=mycomp,dc=com`) respectively groups (`cn=groups,dc=mycomp,dc=com`) is assumed.
- The **user** template is a special case. It assumes a search on objects of type "person" and uses the default search base for users. The other parameters are used in analogy to the search template case; for example (Example 4):
- `Query:                       Peter`
- `Query template:              `**`user`**
- `AdditionalParameter1:  equal`
- `AdditionalParameter2:  uid`
  `AdditionalParameter3:  uid`

  Note: Do not set `AdditionalParameter4` and `AdditionalParameter5`.

- The **usersOfGroup** template is a special case of the search template. It assumes a search on objects of type "group" and uses the default search base of Member Manager for groups. The other parameters are used in analogy to the search template case; for example (Example 5):
- `Query:                       department1`
- `Query template:              `**`usersOfGroup`**
- `AdditionalParameter1:  equal`
- `AdditionalParameter2:  cn`
  `AdditionalParameter3:  uid`

  Note: Do not set `AdditionalParameter4` and `AdditionalParameter5`.

Nobody

Use this verb to deny normal users access to the work item; only the process administrator and the process choreographer system administrator have access. This verb has no parameters and is supported by all plug-in including the Member Manager plug-in.

Person Search

Use this verb to search for people based on an attribute match.

All users are retrieved featuring a specified attribute and a specified value, for example "Profile" equaling "Employee". A process modeler can set a number of parameters when selecting the verb for an activity role:

Parameter	Use	Type	Supported by Member Manager	Description
UserID	Optional	String	Yes	The user ID of the users to retrieve

Parameter	Use	Type	Supported by Member Manager	Description
Profile	Optional	String	Yes	The profile of the users to retrieve
LastName	Optional	String	Yes	The last name of the users to retrieve
FirstName	Optional	String	Yes	The first name of the users to retrieve
MiddleName	Optional	String	Yes	The middle name of the users to retrieve
Email	Optional	String	Yes	The e-mail address of the users to retrieve
Company	Optional	String	Yes	The company to which the users belong
DisplayName	Optional	String	Yes	The display name of the users to retrieve
Secretary	Optional	String	Yes	The secretary of the users to retrieve
Assistant	Optional	String	Yes	The assistant of the users to retrieve
Manager	Optional	String	Yes	The manager of the users to retrieve
Department	Optional	String	Yes	The department to which the users belong
Phone	Optional	String	Yes	The telephone numbers of the users to retrieve
Fax	Optional	String	Yes	The fax number of the users to retrieve
Gender	Optional	String	Yes	Whether the user is male or female
Timezone	Optional	String	Yes	The time zone in which the users are located
PreferredLanguage	Optional	String	Yes	The preferred language of the users to retrieve

One attribute (Type, Industry Type, ...) should be set at a time. Only the first attribute in the list will be evaluated.

The specification of the attribute to be retrieved for the users (i.e. "uid") are fixed in the transformation file used by the plug-in configuration, for example MemberManagerTransformation.xsl located in the *app_server_root*/ProcessChoreographer/Staff directory. If changes are needed, the .xsl file has to be adapted accordingly.

Example (change user return attribute to "cn"):

```
        <xsl:template name="PersonSearch">
        ...
        <swmm:search>
                       <xsl:attribute
name="returnType">person</xsl:attribute>
                       <xsl:attribute
name="returnAttribute">cn</xsl:attribute>
                       <xsl:attribute name="searchAttribute"><xsl:value-
of select="$searchparam"/></xsl:attribute>
                       <xsl:attribute
name="operator">equal</xsl:attribute>
                       <xsl:attribute name="referenceValue"><xsl:value-
of select="staff:parameter"/></xsl:attribute>
                       <xsl:attribute
name="referenceType">String</xsl:attribute>
        </swmm:search>
        </xsl:template>
```

Role Members

Use this verb to retrieve the users associated with a business process role.

The verb translates into a search for all entries of type (i.e. LDAP objectclass) "person" which are located under a predefined search base, for example "o=roleName,cn=role,dc=mycomp" and retrieving a predefined attribute value, for example "uid".

A process modeler can set a number of parameters when selecting the verb for an activity role. Except for the role names, none of the parameters are evaluated during the staff verb resolution.

Parameter	Use	Type	Supported by Member Manager	Description
RoleName	Mandatory	String	Yes	Role name of the users to retrieve
IncludeNestedRoles	Mandatory	Boolean	No	Not evaluated
Domain	Optional	String	No	Not evaluated
AlternativeRoleName1	Optional	String	Yes	An alternative role name for the user
AlternativeRoleName2	Optional	String	Yes	An alternative role

Parameter	Use	Type	Supported by Member Manager	Description
				name for the user

The specification of the search parameters (type, search base, retrieved attribute) is fixed in the transformation file used by the plug-in configuration, for example `MemberManagerTransformation.xsl` located in the *app_server_root*`/ProcessChoreographer/Staff` directory. If changes are needed, the `.xsl` file has to be adapted accordingly. Adaptations are done in analogy to the "Department Members" case (see above).

Users

Use this verb to define staff query for a user whose name is known.

Use full names, for example `"uid=wpsadmin,cn=users,dc=mycomp"` to specify values. A process modeler can set a number of parameters when selecting the verb for an activity role:

Parameter	Use	Type	Supported by Member Manager	Description
Name	Mandatory	String	Yes	The name of the user to retrieve
AlternativeName1	Optional	String	Yes	An alternative user name; use this parameter to retrieve more than one user
AlternativeName2	Optional	String	Yes	An alternative user name; use this parameter to retrieve more than one user

The specification of the attribute to be retrieved for the users (i.e. "uid") are fixed in the transformation file used by the plug-in configuration, for example `MemberManagerTransformation.xsl` located in the *app_server_root*`/ProcessChoreographer/Staff` directory. If changes are needed, the `.xsl` file has to be adapted accordingly.

Example (change user return attribute to "cn"):

```
        <xsl:template name="Users">
        ...
        <swmm:user>
                        <xsl:attribute name="id"><xsl:value-of
select="$username"/></xsl:attribute>
                        <xsl:attribute
name="idType">memberDN</xsl:attribute>
```

```
                       <xsl:attribute
name="attribute">cn</xsl:attribute>
        </swmm:user>
        ...
        </xsl:template>
```

Users by user ID

Use this verb to define a staff query for a user whose user ID is known.

Use short names for example "wpsadmin" to specify values. A process modeler can set a number of parameters when selecting the verb for an activity role:

Parameter	Use	Type	Supported by Member Manager	Description
UserID	Mandatory	String	Yes	The user ID of the user to retrieve
AlternativeID1	Optional	String	Yes	An alternative user ID; use this parameter to retrieve more than one user
AlternativeID2	Optional	String	Yes	An alternative user ID; use this parameter to retrieve more than one user

Users by user ID without Named Users

Use this verb to define a staff query for users whose user ID is known, while excluding explicitly named user IDs. Use short names to specify values, for example, wpsadmin. This verb does not require access to a staff repository.

Parameter	Use	Type	Description
UserID	Mandatory	string	The user ID of the user to retrieve. Supports context variables and custom properties, such as %htm:task.potentialStarters%
AlternativeID1	Optional	string	An additional user ID. Use this parameter to retrieve more than one user.
AlternativeID2	Optional	string	An additional user ID. Use this parameter to retrieve more than one user.
NamedUsers	Mandatory	string	The user IDs of the users to exclude from the user ID list. Supports context

Parameter	Use	Type	Description
			variables and custom properties, such as %wf:activity(...).owner%

8 Resources about Business Process Choreographer staff plug-in providers

The staff plug-in providers concept is defined as part of Process Choreographer, which is a constituent part of the IBM WebSphere Process Server product. For a detailed description of the concept and general usage information, please refer to the following product Information Center and available technical support papers:

- WebSphere Process Server Library at http://www.ibm.com/software/integration/wps/library/
- IBM WebSphere Application Server Enterprise Process Choreographer: Staff Resolution Architecture at http://www.ibm.com/developerworks/websphere/zones/was/wpc.html
- WebSphere Application Server Enterprise Process Choreographer: Programming Model for Staff Resolution at http://www.ibm.com/developerworks/websphere/zones/was/wpc.html
- WebSphere Application Server Enterprise Process Choreographer: Staff Resolution Parameter Reference at http://www.ibm.com/developerworks/websphere/zones/was/wpc.html