SAP Composite Application Framework

Creating a Callable Object in Group: User Management

Version 1.00 – July 2006

SAP AG
Neurottstraße 16
69190 Walldorf
Germany
T +49/18 05/34 34 24
F +49/18 05/34 34 20
www.sap.com

Typographic Conventions

Type Style	Represents
Example Text	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options.
	Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
`Example text`	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
`Example text`	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
`EXAMPLE TEXT`	Keys on the keyboard, for example, F2 or ENTER.

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Contents

Scenario 1

About This Document 4

- General Prerequisites 4
- Applicable Releases 4
- Disclaimer 4

The Step-By-Step Solution 5

- Approaches to Building the Model 5
  - Top-Down Approach 5
  - Bottom-Up Approach 6
- Create a Callable Object: Choose a User 7
- Create a Callable Object: Read User Information 8
- Create a Callable Object: Assign Users to Process Role 10
- Create a Callable Object: Return Process Role Member List 11
- Create a Callable Object: Resolve Manager of User 12
  - Prerequisites 12
  - Creation of Callable Objects 14
- Create a Callable Object: Check User 16
- Finalizing the Callable Objects 17

Scenario

Guided Procedures (GP) provides a set of callable object implementations, which you can use by simply configuring them in the GP Design Time.

There are six types of callable objects collected in the user management group; all of them are centered around user-related functionality.

The table below lists the types and gives a short description of their functionality, parameters and special configuration aspects.

Callable Object	Description
Choose a User	**Type:** Web Dynpro-based form **Functionality:** Provides a user interface for choosing a user from user management. Returns the unique identifier of the user as well as additional data in its output data structure. **Component:** `com.sap.caf.eu.gp.ui.wdco.pickuser.WDCOPickUser` **Application:** `sap.com/caf~eu~gp~ui~wdco` **Output parameters:** 1. `uniqueId` – Unique ID of the user found. 2. `user` – Details about the user found (for example address, phone number, company). **Result states:** `COMPLETED` – Completed successfully.

Read User Information	**Type:** Java Callable Object for background execution **Functionality:** Reads and returns user information based on the unique ID, logon ID or current user. If you enter one of the three options, detailed information about the user is displayed. **Class:** `com.sap.caf.eu.gp.callobj.ume.GetUserInfoCO` **Container:** `caf~eu~gp~actions` **Input parameters:** 1. `UNIQUE_ID` - Unique ID of user. 2. `UNIQUE_NAME` - User name. 3. `LOGON_ID` - Logon ID of user. **Output parameters:** 1. `uniqueid` - Unique ID of the user found. 2. `userdata` – Details about the user found (for example address, phone number, company). **Configuration parameters:** `RESOLUTION_MODE` - Search criteria is the unique ID of the user (for example `USER.PRIVATE_DATASOURCE.un:Guest`), unique user name (for example `Guest`), logon ID, or the ID of the user that is currently logged on. **Result states:** `COMPLETED` – Completed successfully.
Assign Users to Process Role	**Type:** Java Callable Object for background execution **Functionality:** Assigns users to a process role. **Class:** `com.sap.caf.eu.gp.callobj.processrole.ProcessRoleAssignmentUserCO` **Container:** `caf~eu~gp~actions` **Input parameters:** `User_List` – List of the users to be assigned. **Result states:** `manresolve.completed` – Manager resolution completed successfully.

Return Process Role Member List	**Type:** Java Callable Object for background execution **Functionality:** Enables you to get the users assigned to the roles for the current process. **Class:** `com.sap.caf.eu.gp.callobj.processrole.UserListResolutionCO` **Container:** `caf~eu~gp~actions` **Output parameters:** `User_List` – List of users assigned to the roles in the current process. **Result states:** `manresolve.completed` –Manager resolution completed successfully.
Resolve Manager of User (Requires SAP HR)	**Type:** Java Callable Object for background execution **Functionality:** Returns the manager of a specified user using a backend SAP Human Resources system. **Class:** `com.sap.caf.eu.gp.callobj.manager.SAPHRManagerResolveCO` **Container:** `caf~eu~gp~actions` **Input parameters:** `User_List` – User specified **Output parameters:** `Manager_List` – Manager of specified user **Configuration parameters:** Using the configuration parameters of the callable object, you define the endpoint alias for the system where SAP HR is configured. `ept.alias.name` – Endpoint alias name. The name must always be of type Remote Function Call. Furthermore, the portal system alias must be defined. **Result states:** `manresolve.completed` – Manager resolution completed successfully. **Exceptions:** 1. `noactive.plvariant` – No active plan variant is available 2. `eptalias.name.invalid` – Selected endpoint alias is not valid

Check User	**Type:** Java Callable Object for background execution **Functionality:** Checks whether a specified user is unique. This is done by searching the UME by ID or user name for a particular user. If the search returns multiple results, they are displayed as a structure. **Class:** com.sap.caf.eu.gp.callobj.bckgd.checkuser.CheckUser **Container:** `caf~eu~gp~actions` **Input parameters:** 1. `input.user.id` – User ID 2. `input.user.name` – User name **Output parameters:** 1. `output.user.id` – Unique ID of the user found 2. `output.users` – List of the users found **Result states:** 1. `ONE_USER` – A single user matches the search criteria 2. `ZERO_USERS` – No users match the search criteria 3. `MANY_USERS` – Multiple users match the search criteria

About This Document

This document describes how to create the six types of callable objects in the user management group.

General Prerequisites

An SAP HR System should be configured at the Enterprise Portal for the callable object *Resolve Manager of User*.

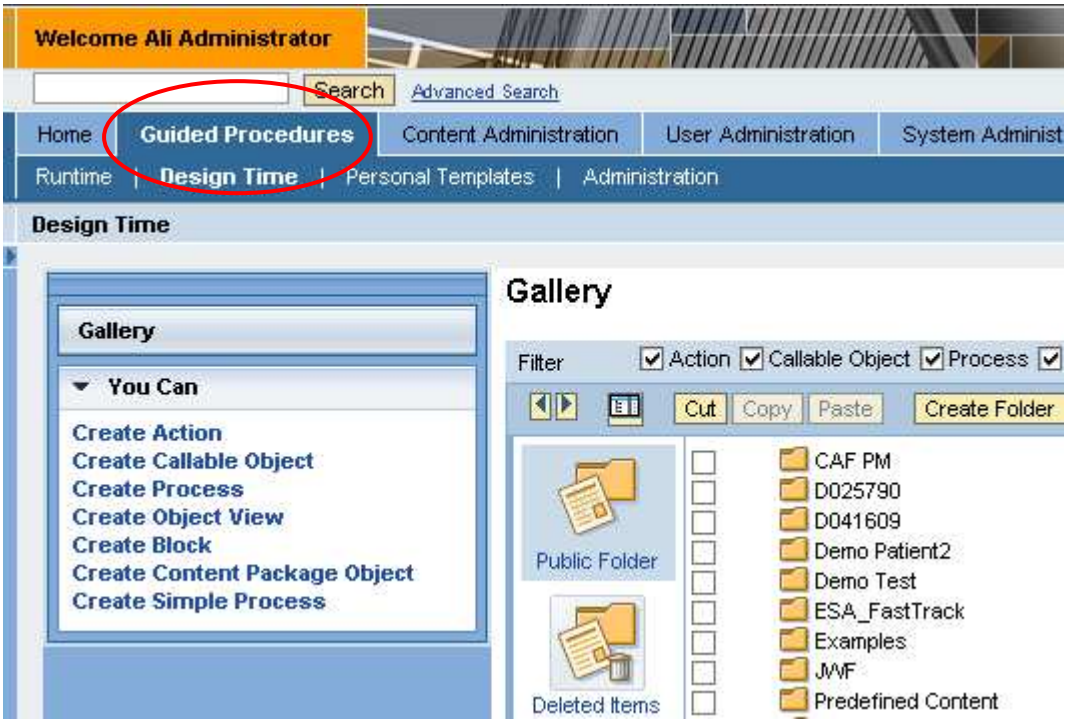
Applicable Releases

This tutorial is compatible with the following release “Beginning with SAP NetWeaver ’04s”.

Disclaimer

The Step-By-Step Solution

Open your SAP Enterprise Portal with http://<Server>:<Port>/irj/portal, navigate to tab **Guided Procedures** and to tab **Design Time**.



Approaches to Building the Model

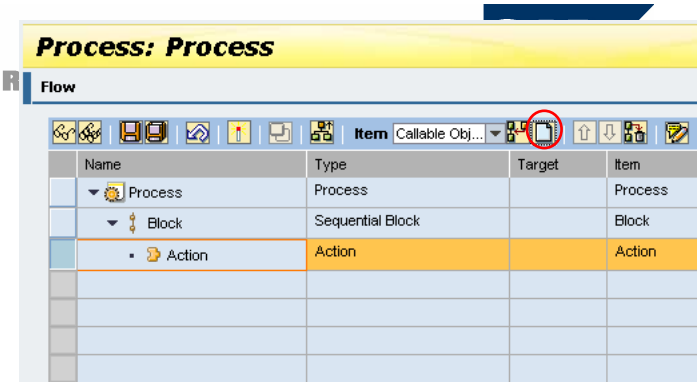
You can choose the top-down approach to create the process, block, and action and then insert a new callable object; alternatively you can use the bottom-up approach to start with the callable object without having any other model item.

Top-Down Approach

In this case you create all the design time objects (process, block, and action) that will make use of this callable object beforehand.

For further details on this top-down approach see **Error! Reference source not found.**.

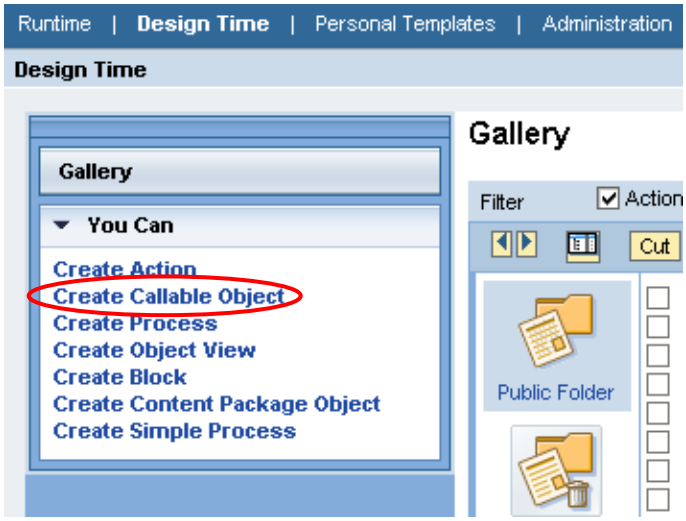
1. Click on the **Create New** icon to embed a new callable object in the existing action.



Bottom-Up Approach

In this case you create the callable object first and insert it in the embedding Design Time objects later.

1. On the left side of window 'You Can', select **Create Callable Object** to open the Callable Object Design Time



Create a Callable Object: Choose a User

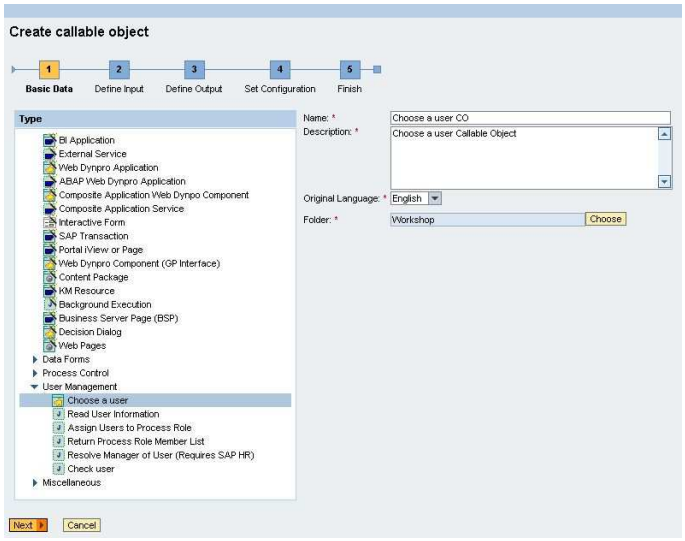
This callable object provides a user interface for choosing a user from user management. It returns the unique identifier of the user as well as additional data in its output data structure.

1. Select the type of callable object: **Choose a user**.

Enter the basic data for the callable object:

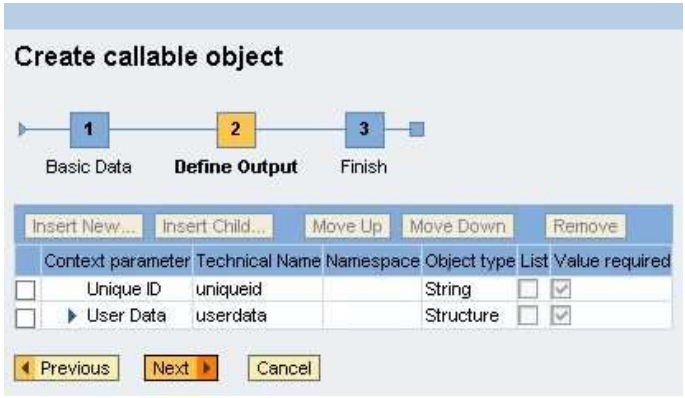
- Name
- Description
- Language
- Location (Click **Choose** to select the folder for the location)

Click **Next** the go to the next screen.



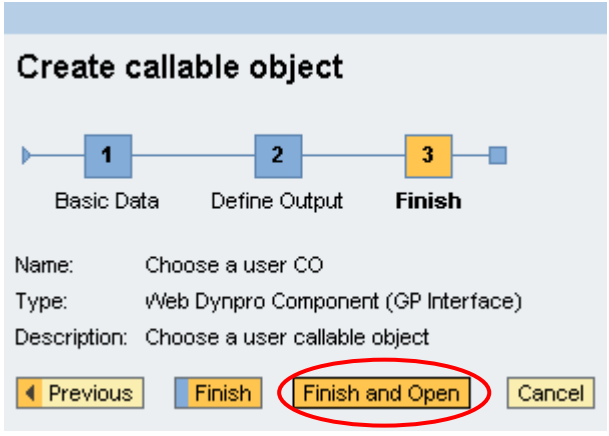
2. As you can see, the output parameters are pre-defined.

Click **Next** to continue.



3. Click **Finish and Open**.

Go to step **0** to finish the tutorial.



Create a Callable Object: Read User Information

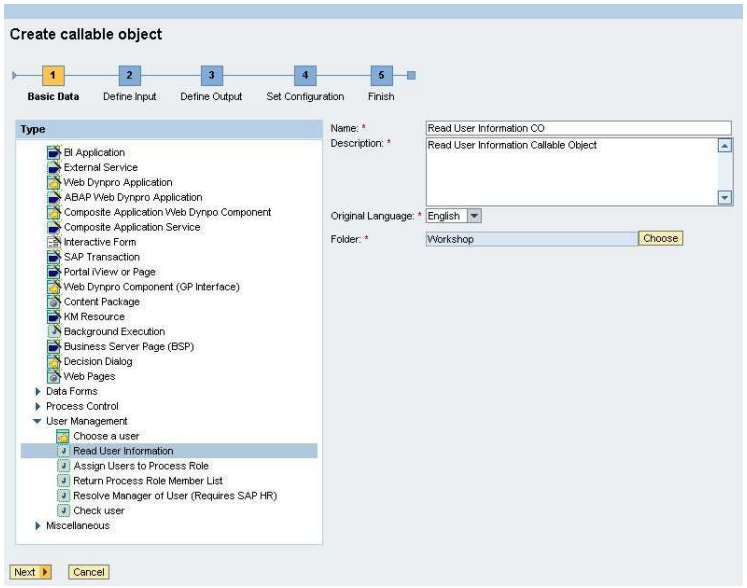
This callable object reads and returns user information based on the unique ID, unique name, the logon ID or returns information about the current user. If you enter one of the four options, detailed information about the user is displayed.

1. In the first screen select the type of callable object: **Read User Information**.

Enter the basic data for the callable object:

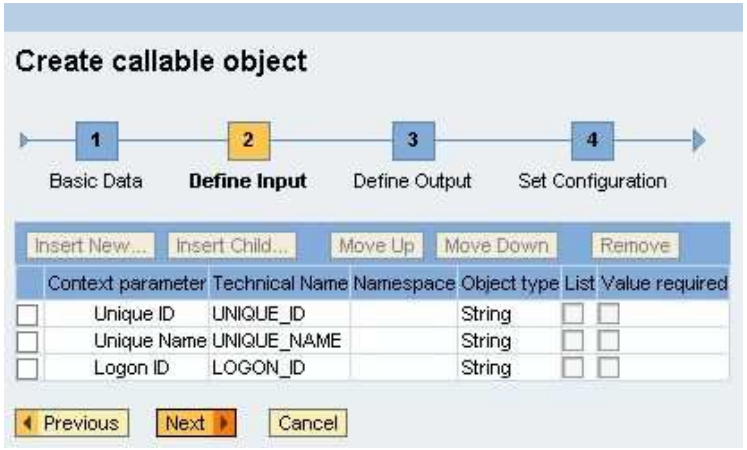
- Name
- Description
- Language
- Location (Click **Choose** to select the folder for the location)

Click **Next.**



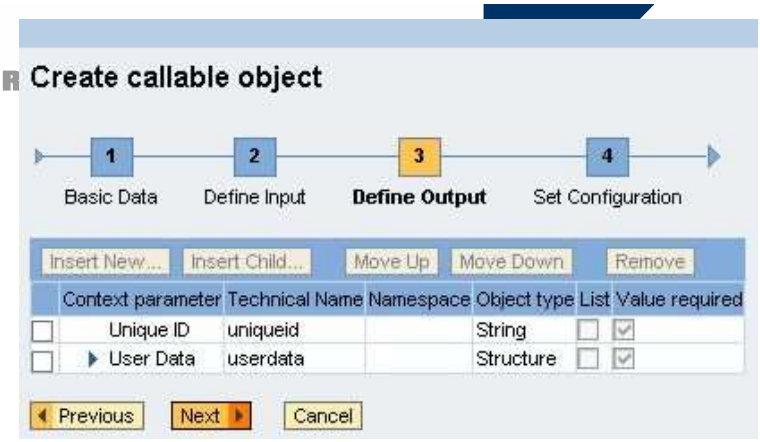
2. The input parameters are displayed.

Click **Next** the go to the next screen.



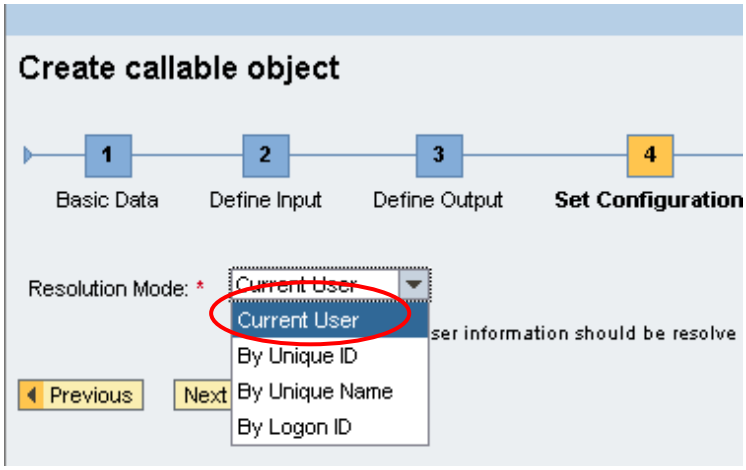
3. The output parameters are displayed.

Click **Next**.



4. In this *Set Configuration* step you can define how the user details should be retrieved. Open the dropdown box *Resolution Mode* and choose whether you want to define the user *By Unique ID*, *By Unique Name, By Logon ID* or you want to get the details of the *Current User*.

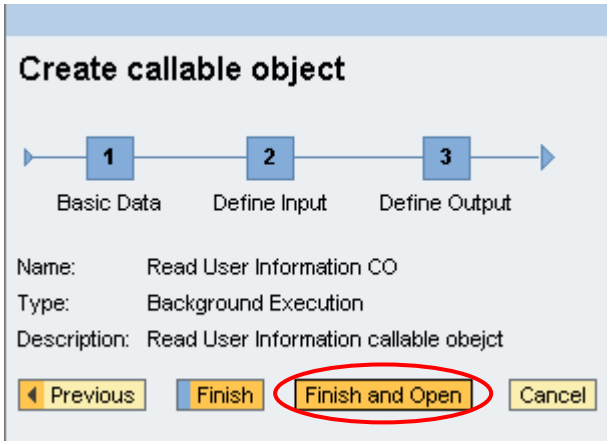
Click **Next.**



5. Click **Finish and Open**.

Go to step **0** to finish the tutorial.

Note for testing: If the user could not be found, the output parameters in the test result screen will be empty although the state result is green.



Create a Callable Object: Assign Users to Process Role

With this callable object you can assign users to the possible roles of the current process.

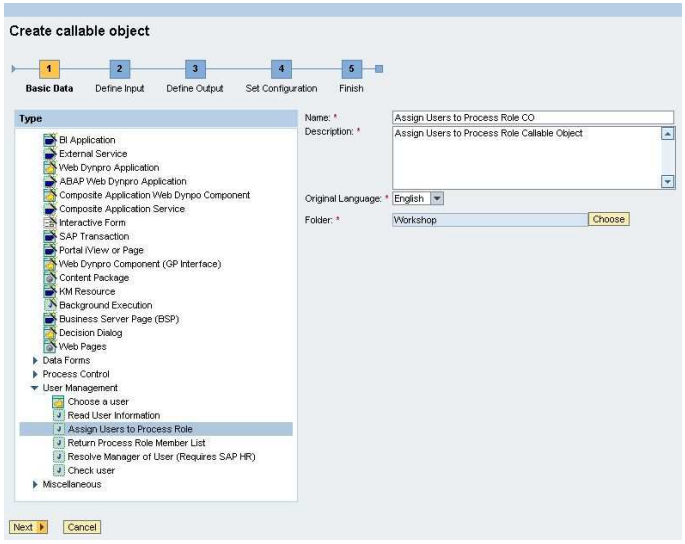
Prerequisites: In order to test this CO it is **necessary** to **integrate it into a process**. The role that is created for the relevant action is filled during runtime with the appropriate data of all the users defined as input parameters of the callable object.

1. Select the type of callable object: **Assign Users to Process Role**.

Enter the basic data for the callable object:

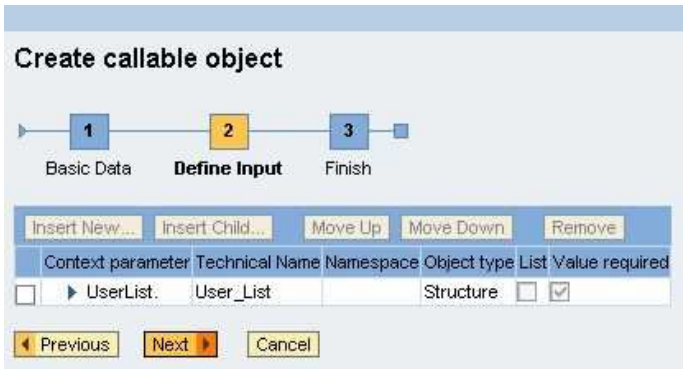
- Name
- Description
- Language
- Location (Click **Choose** to select the folder for the location)

Click **Next** the go to the next screen.



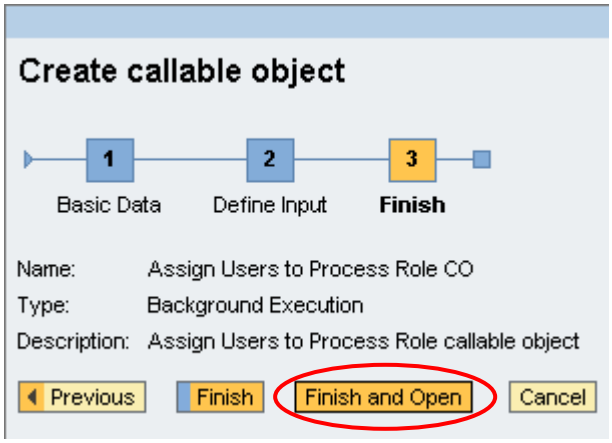
2. Enter a list of users that can be assigned a role at runtime.

Click **Next**.



3. Click on **Finish and Open**.

Go to step **0** to finish the tutorial.



Create a Callable Object: Return Process Role Member List

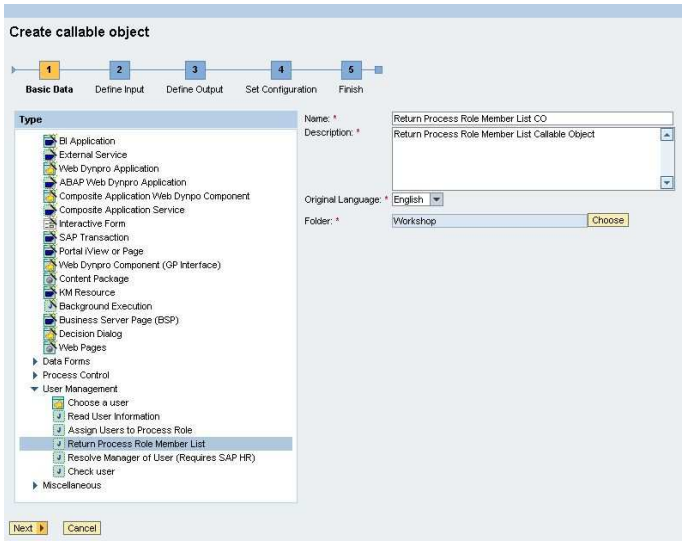
This callable object enables you to get the users assigned to the roles of the current process.

1. Select the type of callable object: **Return Process Role Member List**.

Enter the basic data for the callable object:

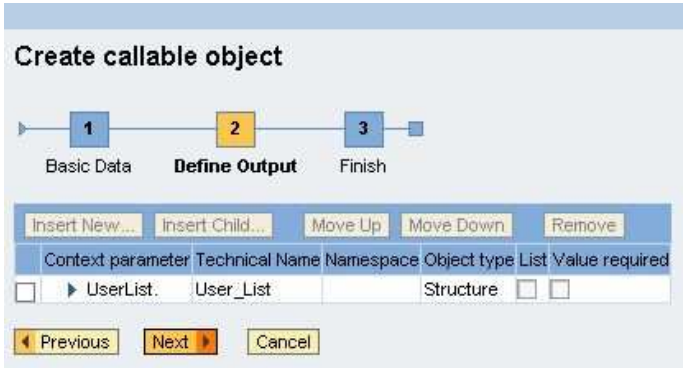
- Name
- Description
- Language
- Location (Click **Choose** to select the folder for the location)

Click **Next**.



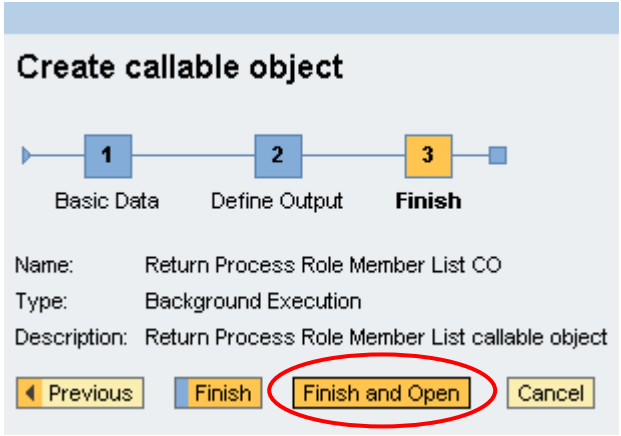
2. The list of users assigned to the roles of the process is displayed.

Click on **Next**.



3. Click on **Finish and Open**.

Go to step **0** to finish the tutorial.



Create a Callable Object: Resolve Manager of User

This callable object returns the manager of a specified user using a backend SAP Human Resources system.

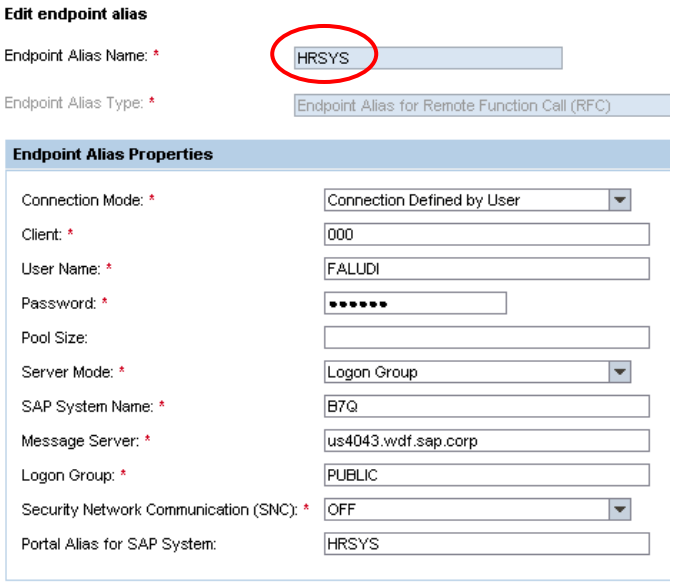
Prerequisites

This callable object needs an SAP HR System and a valid user for that system.

1. Configure an end point in GP (*Guided Procedures → Administration*) representing the SAP HR System – it should be an RFC alias.

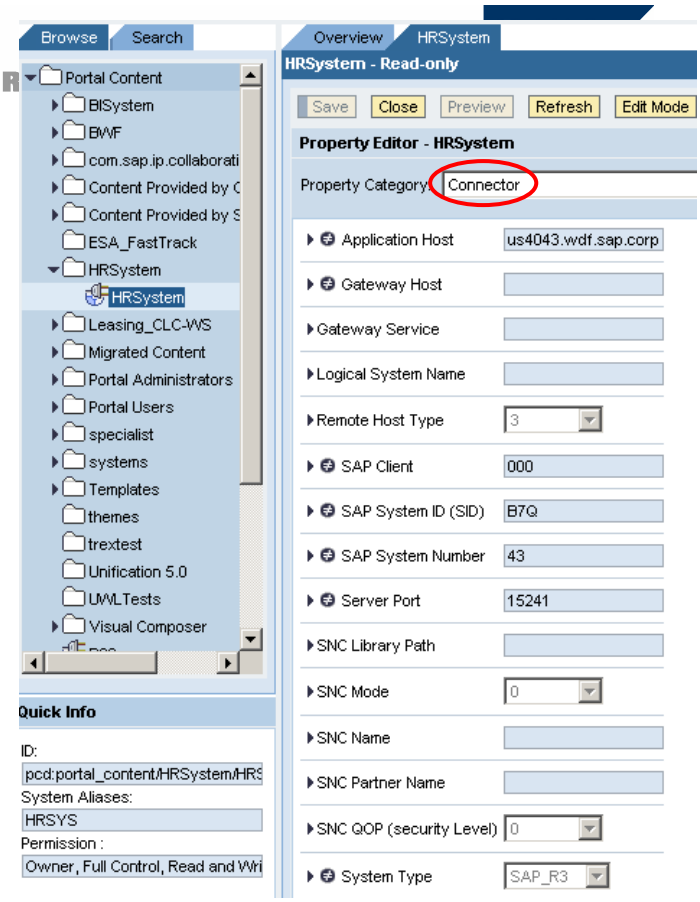
   In the example, the alias is "*HRSYS*". A name and password is entered for the user in that system. It is important to enter an alias for the *Portal Alias for SAP System* field (it can be the same as the end point alias) and it should be the alias of the system added to the portal system landscape (see next step).

   (Test it before saving to make sure that the settings are correct.)



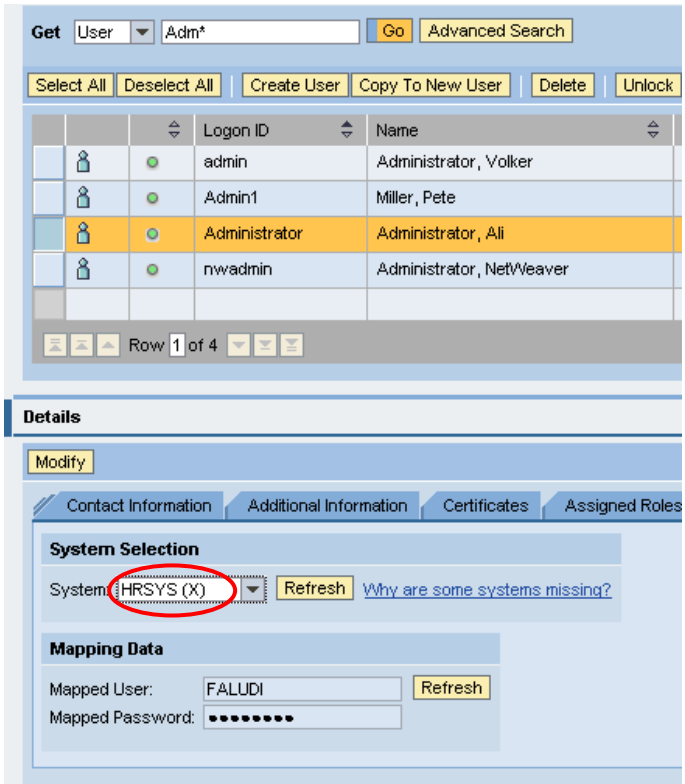
2. The SAP HR System must be added to the portal system landscape with the configured connector-related parameters.

The alias must be the same as is set for the GP end point (see previous step).

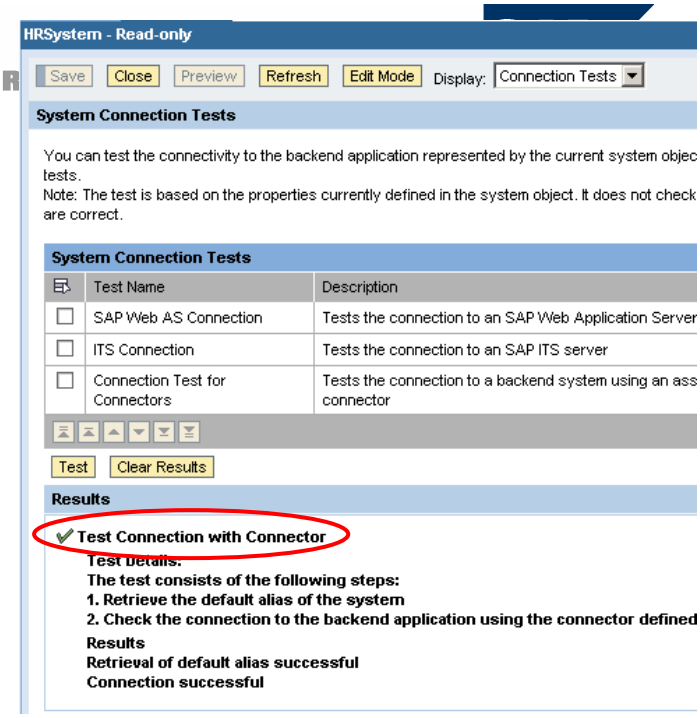


3. Define the user mapping in the portal following the path *User Administration → Identity Management*).

Map your portal user to your user in the SAP HR System.



4. Test the connection of the HR System in the portal landscape – apply only the *Connection Test for Connectors*.



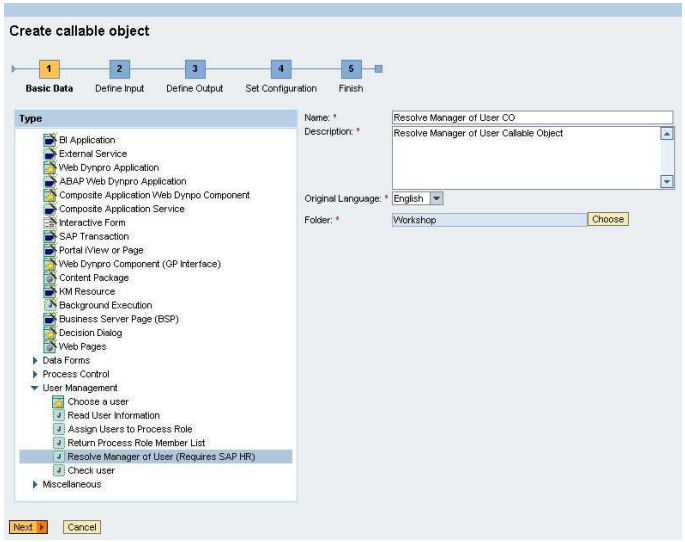
Creation of Callable Objects

1. Select the type of callable object: **Resolve Manager of User**.

Enter the basic data for the callable object:

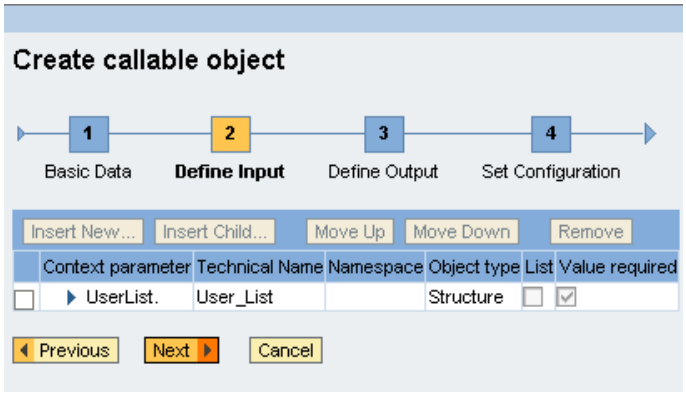
- Name
- Description
- Language
- Location (Click **Choose** to select the folder for the location)

Click **Next**.



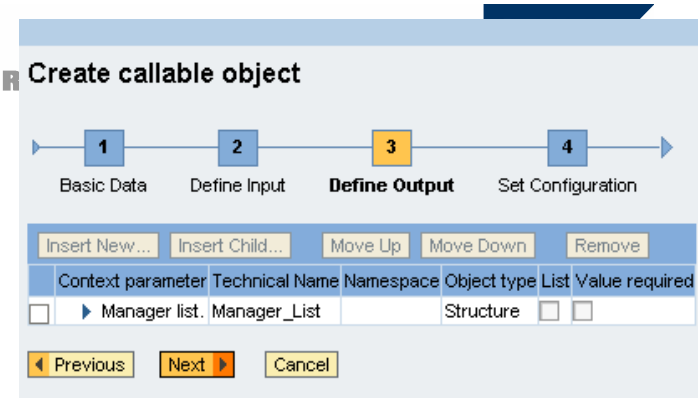
2. Enter a list of users.

Click **Next**.



3. A list of relevant managers is displayed.

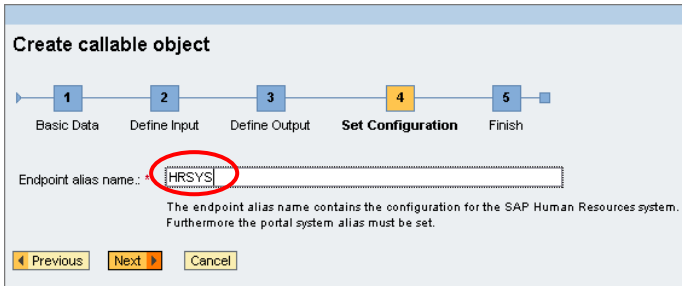
Click **Next**.



4. In this *Set Configuration* step, enter the alias of the HR System that is used as a back-end.

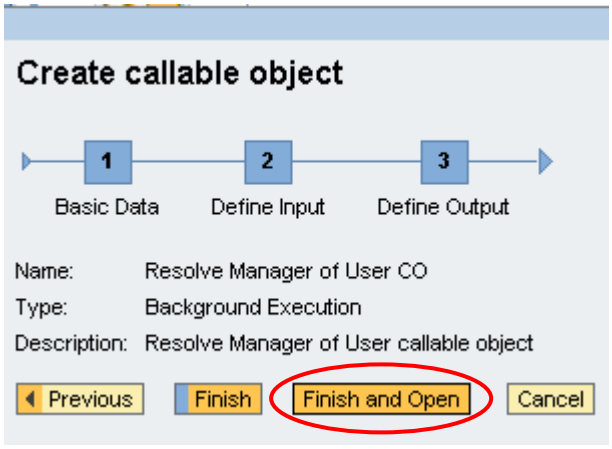
It is "*HRSYS*" in the example.

Click **Next**.



5. Click on **Finish and Open**.

Go to step **0** to finish the tutorial.



Create a Callable Object: Check User

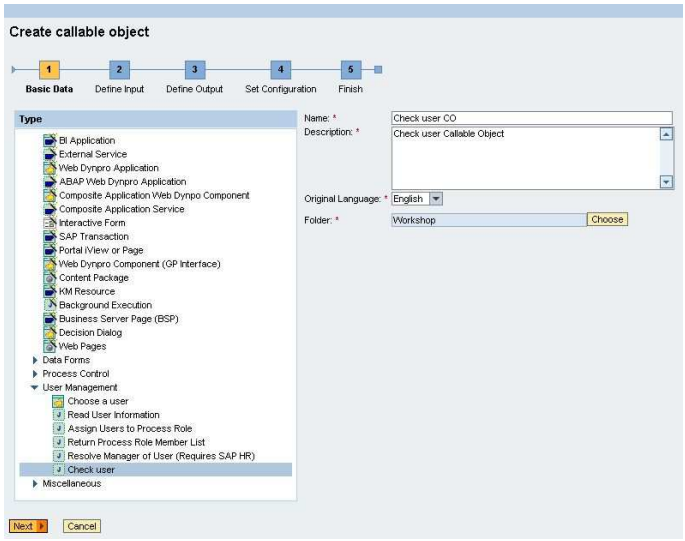
This callable object checks whether a specified user is unique. This is done by searching the UME by ID or user name for a particular user. If the search returns multiple results, they are displayed as a structure.

1. Select the type of callable object: **Check user**.

Enter the basic data for the callable object:

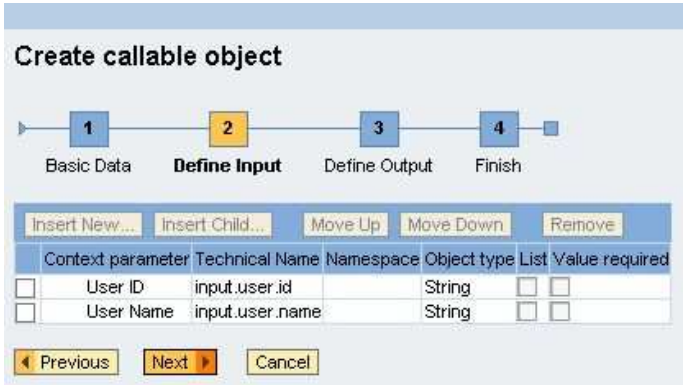
- Name
- Description
- Language
- Location (Click **Choose** to select the folder for the location)

Click **Next**.



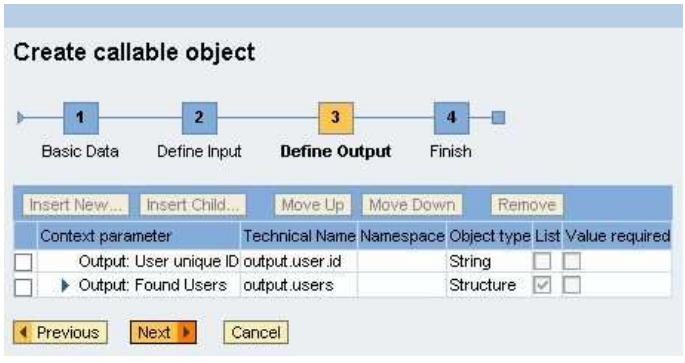
2. Enter *User ID* or *User Name* depending on the configuration of this CO.

Click **Next**.

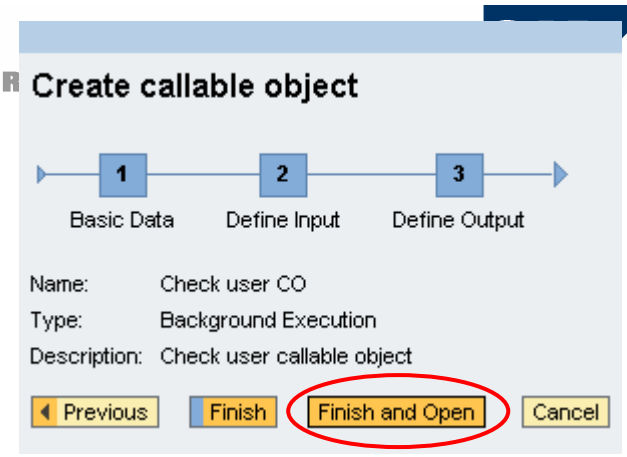


3. The unique user ID and the structure of users that matches the input criteria is displayed.

Click **Next.**



4. Click **Finish and Open**.



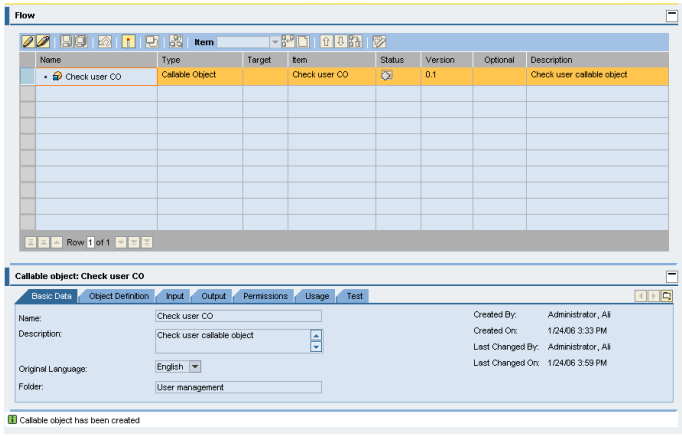
Go to step **0** to finish the tutorial.

Finalizing the Callable Objects

1. After clicking Finish and Open, the page where you can maintain your callable object is displayed.

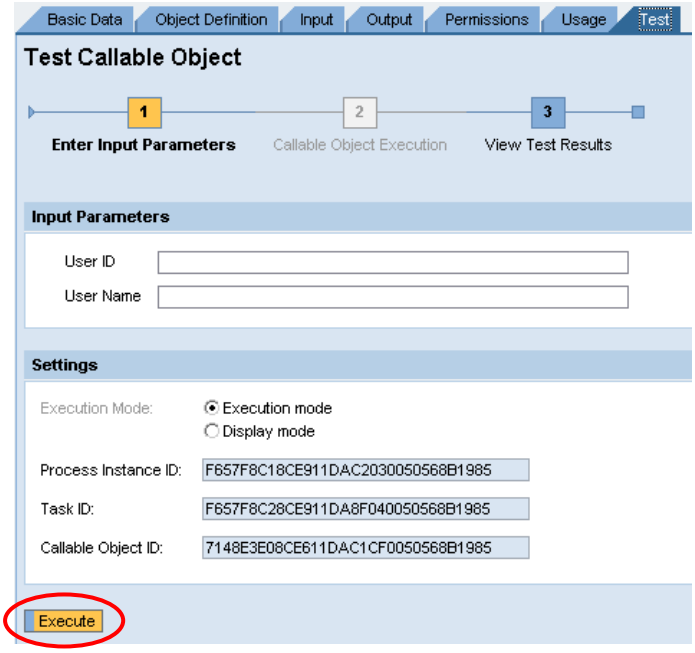
   Click the **Test** tab at the bottom of the page.

   .

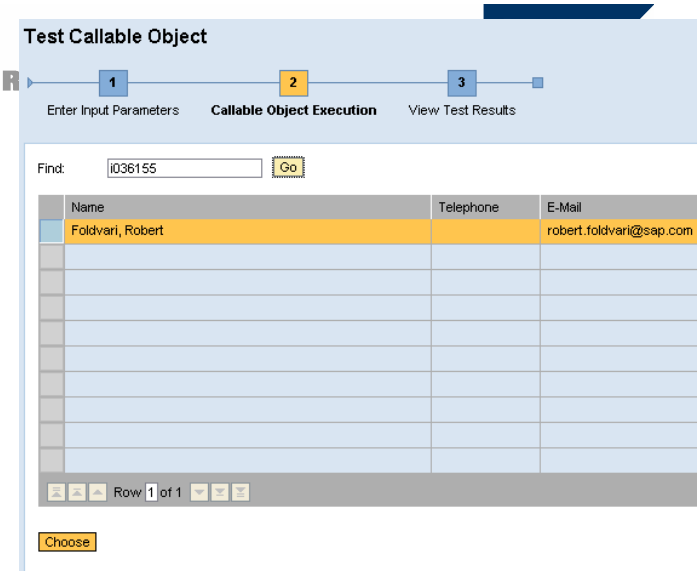


2. If there are any input parameters for the callable object, enter them.

   Click **Execute**.



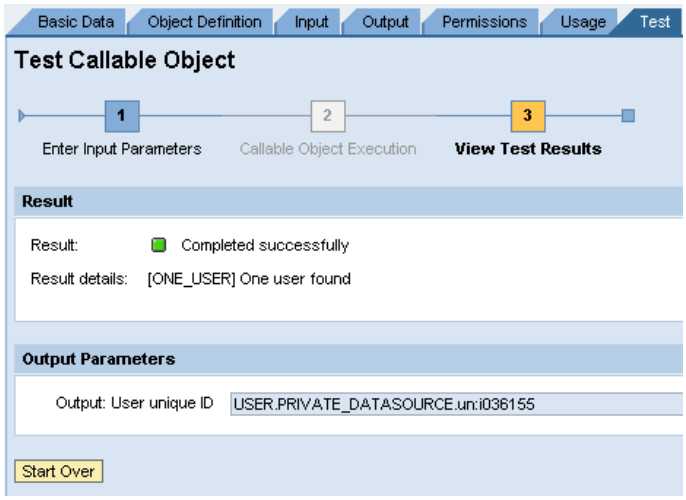
3. There is only a user interface for the *Callable Object Execution* step for the type *Choose a user*.



It provides a Web Dynpro interface for selecting a particular user.

Hint: The wildcard '*' cannot be used; the search only finds exact matches.

4. The callable object was executed and the test results are displayed here.



If the execution was successful you can see the *Result details* – this is the result state described in the table in chapter **Error! Reference source not found.**.

The *Output Parameters* are also displayed.

5. Activate your callable object by clicking the **Activate** icon on the top of the page. Alternatively you can activate the process in which this service method is used.