HP OO-SAP Integration

For the Windows operating system

Software Version: 1.00.00

SAP Integration Guide

Document Release Date: June 2012

Software Release Date: June 2012

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

Trademark Notices

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at:

http://www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Contents

Chapter 1

SAP Integration Introduction

This chapter includes:

Purpose of the SAP Integration

This integration enables administrators to create HP Operations Orchestration (OO) flows that are integrated with SAP.

To learn how to create OO flows, see Studio Guide to Authoring Operations Orchestration Flows in the documentation set for the current OO release,

This document explains how this integration has been implemented, and how the integration's operations and flows communicate between OO and SAP.

Audience

This guide is intended for system administrators who establish and maintain the implementation of the integration between SAP and HP OO. This guide assumes that you have administrative access to both systems.

Prerequisites

To use this integration successfully, you should have administrator-level knowledge of the SAP, the SAP BAPIs, and RFCs.

Supported Versions

Operations Orchestration Version	Operations Orchestration Studio Version	SAP Versions
OO Content Pack 8	9.0 and later	SAP ECC 6.0

Chapter 2

Getting Started

This chapter includes:

What You Need to Know About SAP Before Using the Integration

SAP helps companies of all sizes and industries run better. From back office to boardroom, warehouse to storefront, desktop to mobile devices, SAP empowers people and organizations to work together more efficiently to stay ahead of the competition. This is accomplished by extending the availability of software across on-premise installations, on-demand deployments, and mobile devices.

SAP Use Cases

Following are the major use cases for the SAP integration, and the operations and flows that you can use to implement them.

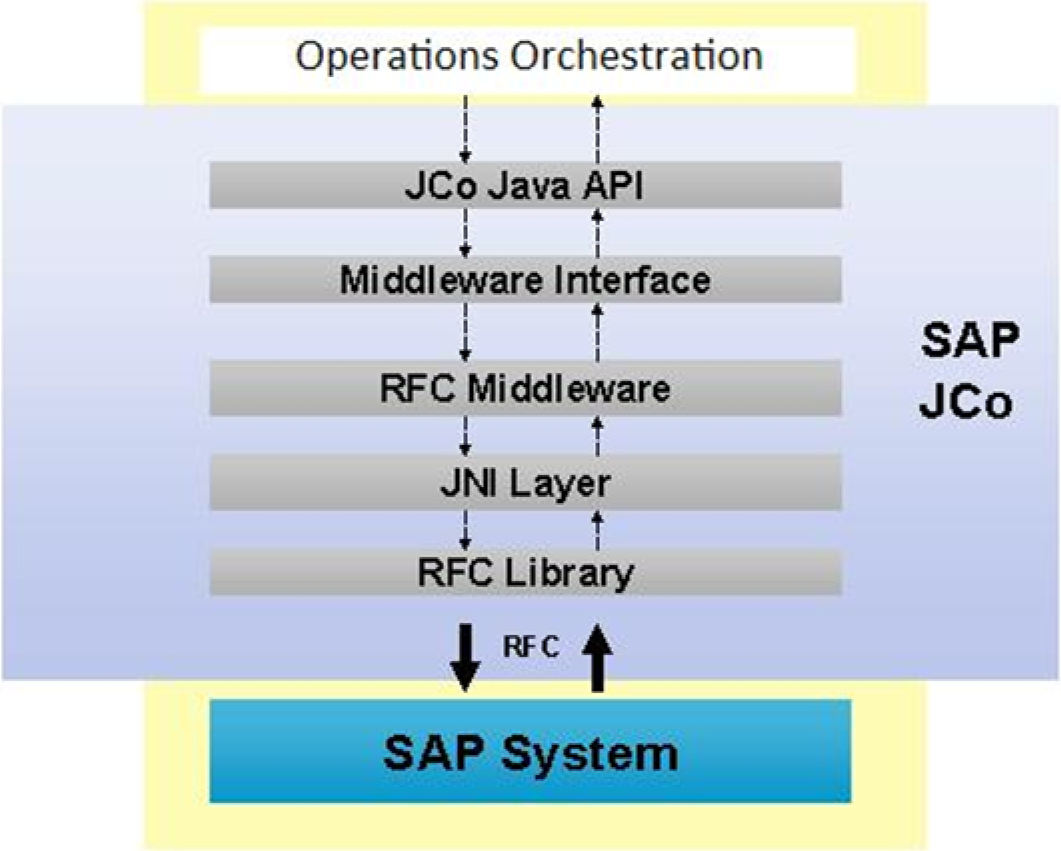
- Execute a Remote Function Call that invokes any given BAPI:
  - Invoke SAP BAPI
- Execute a Remote Function Call that invokes a specific BAPI:
  - BapiObjectMethod (for example, BapiUserGetlist).

OO — SAP Integration Architecture

The OO SAP Integration uses RFCs (Remote Function Calls) to enable the communication between a SAP system and Operations Orchestration. RFC is the standard SAP interface for communicating with SAP systems by calling a function to be executed on a remote system. The function call is based on synchronous communication, meaning that the SAP system must be available at the time the call is made in order to be executed successfully.

Remote Function Calls can be used to invoke BAPIs (Business Application Programming Interfaces). Stored in the Business Object Repository, BAPIs are methods of SAP business object types that provide access to processes and data in business application systems.

To interact with the SAP system and execute specific BAPIs, the SAP Java Connector is used as a middleware component. In the OO – SAP Integration, the SAP Java Connector supports communication with the SAP Server, as shown in the figure below.



SAP Terminology

The following terms are used in the SAP Integration Guide:

BAPI. BAPIs (Business Application Programming Interfaces) are defined as methods of SAP business object types. The BAPIs are standardized with full stability guarantees for their content and interface, to provide access to processes and data in the SAP System.

RFC. RFC (Remote Function Call) is the standard SAP interface for communication between SAP systems. RFC calls a function to be executed on a remote system.

Function module. A procedure that is defined in special ABAP programs, that plays an important role in interaction between SAP systems and remote systems through remote communication.

Business Object. A business object is a general category for data (for example, employee, order, purchase requisition, or organizational unit).

Method. A clearly defined interface that is revealed to the outside so that applications can access the Business Object data. When a method is executed on a Business Object, it can change the object’s internal state, that is, the object’s data (for example, one method that you can use on the Business Object “Employee” is to “check for the employee’s existence”).

BOR. The Business Object Repository is the repository that contains all SAP business object types and SAP interface types as well as their components, such as methods, attributes, and events.

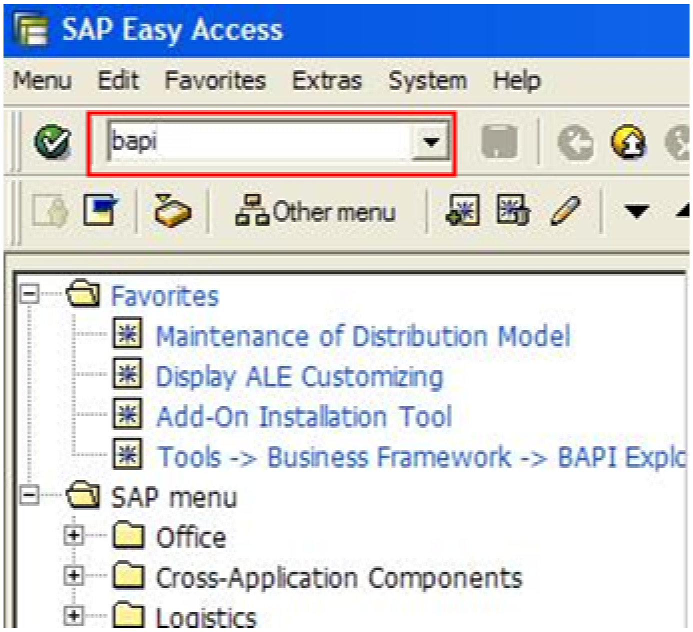
Access the Available BAPIs in the SAP System

To access the BAPIs available in the SAP system:

1. In the SAP Easy Access window, browse to **SAP menu/Tools/Business Framework/BAPI Explorer**

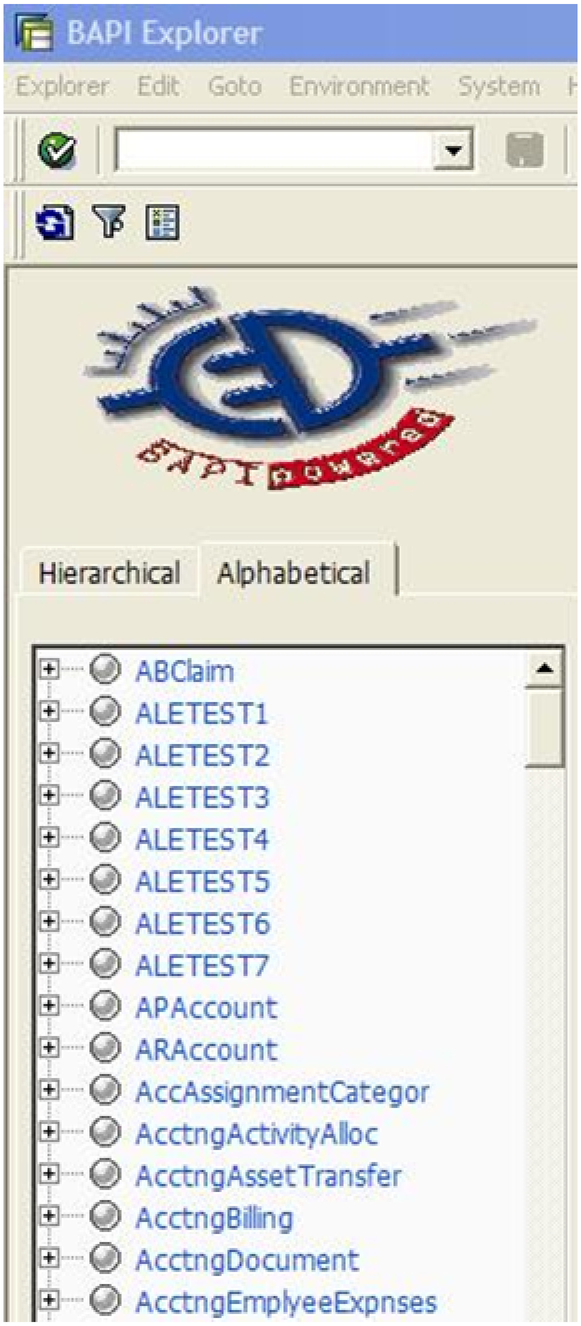
   Or

   Enter the associated transaction code.

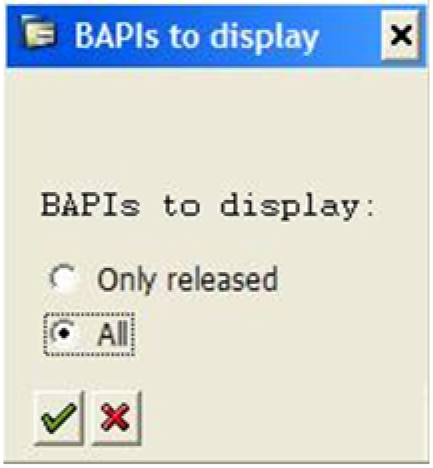

SAP Easy Access
Menu Edit Favorites Extras System Help
bapi
Other menu
Favorites
Maintenance of Distribution Model
Display ALE Customizing
Add-On Installation Tool
Tools -> Business Framework -> BAPI Explc
SAP menu
Office
Cross-Application Components
Logistics

   The list of BAPIs is displayed in a hierarchical view.

2. Click the **Alphabetical** tab to view the BAPIs and their associated Business Objects. The list is sorted alphabetically according to the available objects.



3. Click the button to view a full or partial list of BAPIs. In the BAPIs to display window, select **All** to display all BAPIs and **Only released** (the default) to display only the released BAPIs.



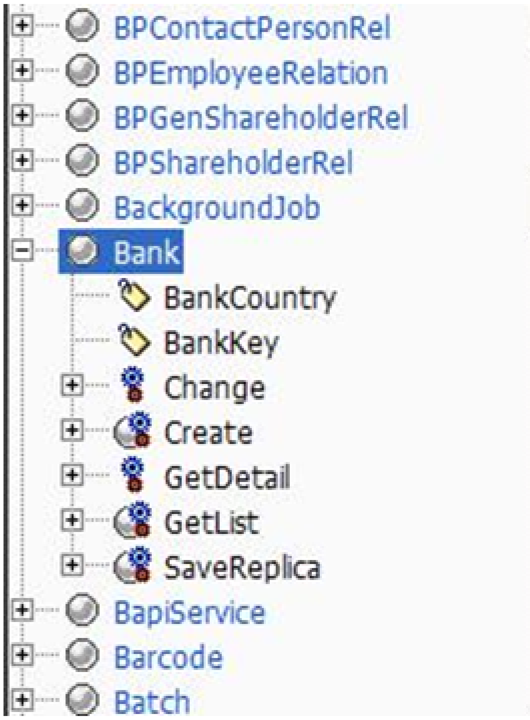
BAPIs to display

BAPIs to display:

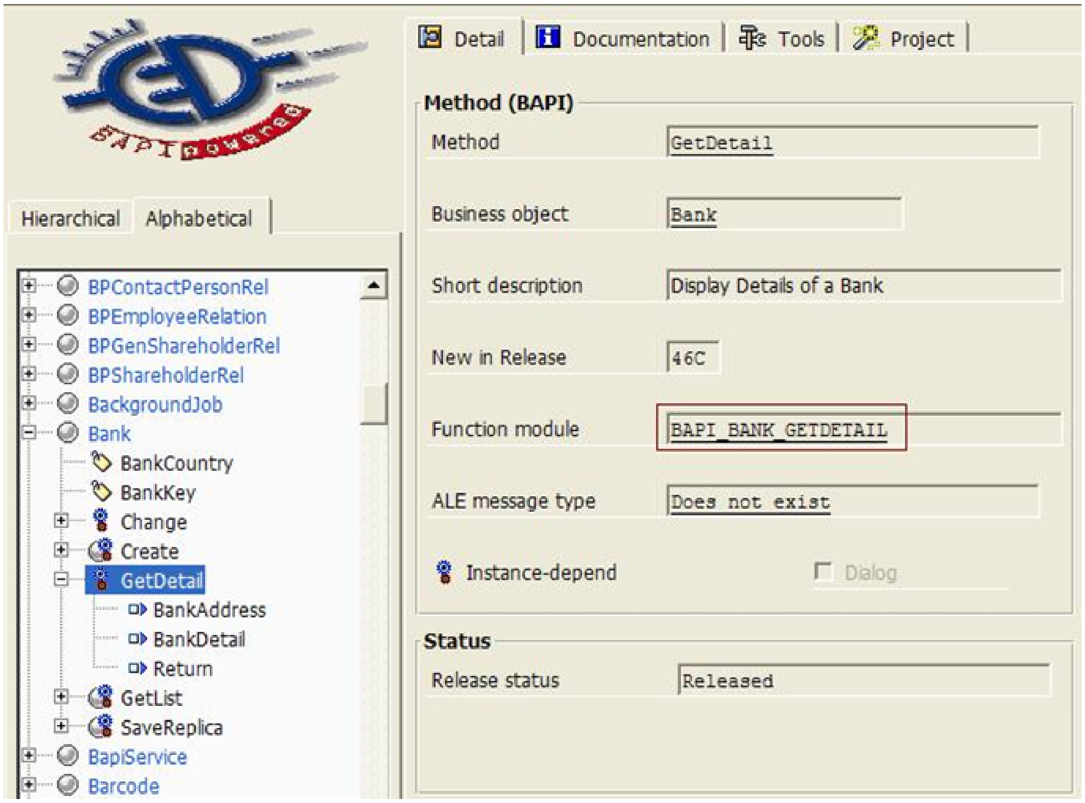
○ Only released

● All

4. Expand the required Business Object in the list to view its associated methods.



5. Select the required method to view its **Function module** value in the **Detail** tab. This is the value of the BAPI parameter used in the OO - SAP Integration.



Method (BAPI)

Method: GetDetail

Business object: Bank

Short description: Display Details of a Bank

New in Release: 46C

Function module: BAPI_BANK_GETDETAIL

ALE message type: Does not exist

Instance-depend

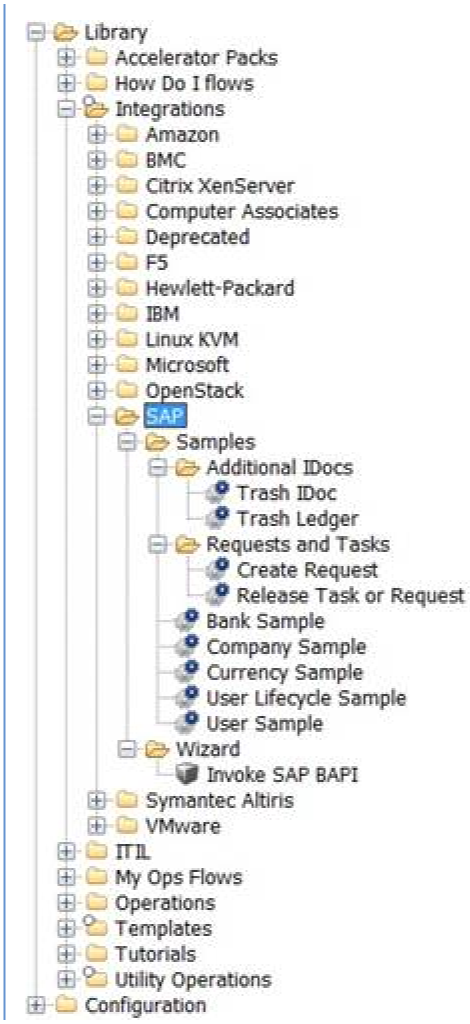
□ Dialog

Status

Release status: Released

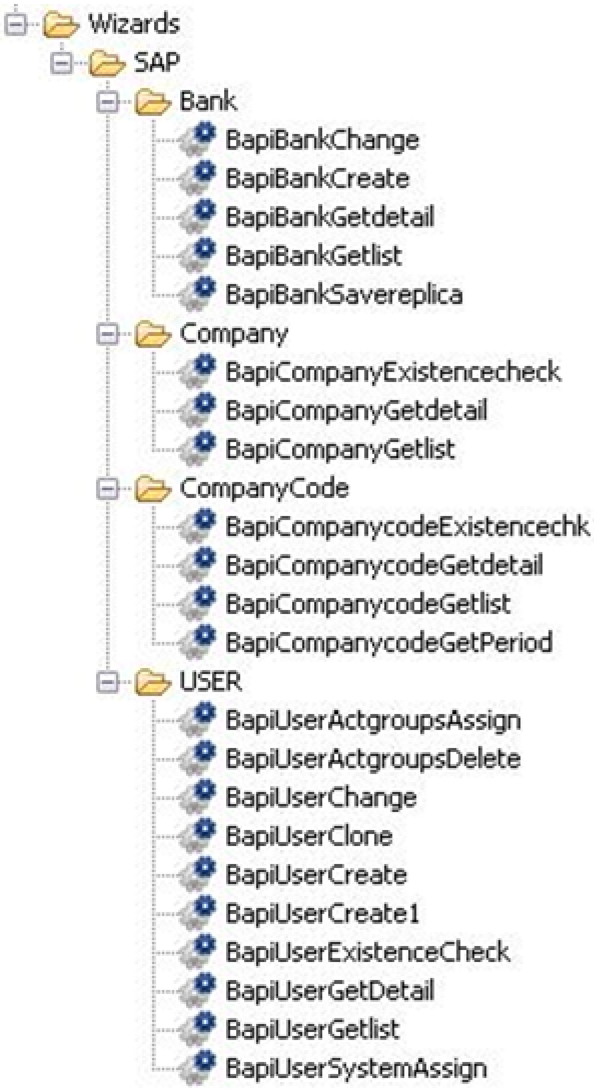
Location of SAP Integration Operations and Flows in OO Studio

The SAP integration includes the generic Invoke SAP BAPI operation and sample flows in the OO Studio **Library/Integrations/SAP/** folder.



```
Library
    Accelerator Packs
    How Do I flows
    Integrations
        Amazon
        BMC
        Citrix XenServer
        Computer Associates
        Deprecated
        F5
        Hewlett-Packard
        IBM
        Linux KVM
        Microsoft
        OpenStack
        SAP
            Samples
                Additional IDocs
                    Trash IDoc
                    Trash Ledger
                Requests and Tasks
                    Create Request
                    Release Task or Request
                Bank Sample
                Company Sample
                Currency Sample
                User Lifecycle Sample
                User Sample
            Wizard
                Invoke SAP BAPI
        Symantec Altiris
        VMware
    ITIL
    My Ops Flows
    Operations
    Templates
    Tutorials
    Utility Operations
Configuration
```

To create customized flows for BAPIs, you must run the SAP Wizard. After running the SAP Wizard and generating customized flows for the requested BAPIs, you must add the created flows to OO Studio by importing the repository that contains them. The generated flows are placed under the folder that you select when running the wizard. The default location where the flows will be placed is under the **Library/Wizards/SAP/** folder, but you have the option of selecting a different library location.



- Wizards
  - SAP
    - Bank
      - BapiBankChange
      - BapiBankCreate
      - BapiBankGetdetail
      - BapiBankGetlist
      - BapiBankSavereplica
    - Company
      - BapiCompanyExistencecheck
      - BapiCompanyGetdetail
      - BapiCompanyGetlist
    - CompanyCode
      - BapiCompanycodeExistencechk
      - BapiCompanycodeGetdetail
      - BapiCompanycodeGetlist
      - BapiCompanycodeGetPeriod
    - USER
      - BapiUserActgroupsAssign
      - BapiUserActgroupsDelete
      - BapiUserChange
      - BapiUserClone
      - BapiUserCreate
      - BapiUserCreate1
      - BapiUserExistenceCheck
      - BapiUserGetDetail
      - BapiUserGetlist
      - BapiUserSystemAssign

Chapter 3

Installation Instructions

This chapter includes:

OO SAP Content Installation Instructions

Before using the SAP Wizard, you must install the OO SAP content, that makes the SAP Integration generic operation and sample flows available in Studio.

Installing OO-SAP Content Manually

If you are installing OO-SAP Content manually, use one of the following methods.

> **Note:** In the following procedures, when you run **OO-SAP-01_00_00-ContentInstaller.jar**, you will see errors about being unable to update the RAS. The installer will retry, and the installation will continue. As long as you complete the procedure, you can ignore these messages.

Installing OO-SAP Content on a Local Central Server

1. Close Studio and make sure that Central and your RAS are running by checking the status of the **RSCentral** and **RSJRAS** services.
2. Ensure that you have OO Content Pack 8 Installed.
3. Download the distributable from http://support.openview.hp.com/selfsolve/patches.
4. Navigate to and copy the OO-SAP Content Installer version of **OO-SAP-01_00_00-ContentInstaller.jar** to the OO home directory in the **\jre1.6\bin\** folder.
5. Click the **Start** menu, and then click **Run**.

6. In the Run dialog box, type cmd and then click **OK**.
7. In the command window, change to the OO home directory in the **\jre1.6\bin\** folder, and then type: `java -jar OO-SAP-01_00_00-ContentInstaller.jar -centralPassword <your Central password>`.
8. Restart the RAS service on all RASes in your OO deployment.

Note the following:

- On Windows 2008 and Windows 2008 R2 servers, you must have administrative privileges to install OO-SAP Content. To get administrative privileges, follow these steps:
  a. On the **Start** menu select **All Programs**, and then **Accessories**.
  b. Right-click **Command Prompt** and then select the **Run as administrator** option.
- If you are using a Central username other than the default admin, use the **centralUsername** parameter in the command shown in step 7 of the previous procedure as follows:

  ```
  java -jar OO-SAP-01_00_00-ContentInstaller.jar --centralUsername
  <nondefault Central username> -centralPassword <your Central
  password>
  ```

  For example:

  ```
  java -jar OO-SAP-01_00_00-ContentInstaller.jar --centralUsername
  sysadmin -centralPassword iconclude
  ```

- If you are installing OO-SAP Content on a non-default port instead of the default port **https://localhost:8443**, use the <centralURL> parameter in the command shown in step 7 of the previous procedure as follows:

  ```
  java -jar OO-SAP-01_00_00-ContentInstaller.jar --centralURL <non-
  default Central URL> -centralPassword <your Central password>
  ```

  For example:

  ```
  java -jar OO-SAP-01_00_00-ContentInstaller.jar --centralURL
  https://central_server1:8080 -centralPassword iconclude
  ```

- To install OO-SAP Content even if there are conflicts between versions of operations, use the **forceInstall** parameter in the command shown in step 7of the previous procedure as follows:

  ```
  java -jar OO-SAP-01_00_00-ContentInstaller.jar  --forceInstall -
  centralPassword<your Central password>
  ```

  For example

  ```
  java -jar OO-SAP-01_00_00-ContentInstaller.jar  --forceInstall -
  centralPassword iconclude
  ```

Installing OO-SAP Content on a Remote Central

1. Make sure that Central and your RAS are running.
2. Ensure that you have OO Content Pack 8 Installed.
3. Download the distributable from http://support.openview.hp.com/selfsolve/patches.

4. Navigate to and copy the OO-SAP Content Installer version of **OO-SAP-01_00_00-ContentInstaller.jar** to a temporary directory.

5. Click the **Start** menu, and then click **Run**.

6. In the Run dialog box, type `cmd` and then click **OK**.

7. In the command window, change to the <temporary directory>, and then type:

   ```
   java -jar OO-SAP-01_00_00-ContentInstaller.jar --centralURL <remote
   Central URL> -centralPassword <your Central password>
   ```

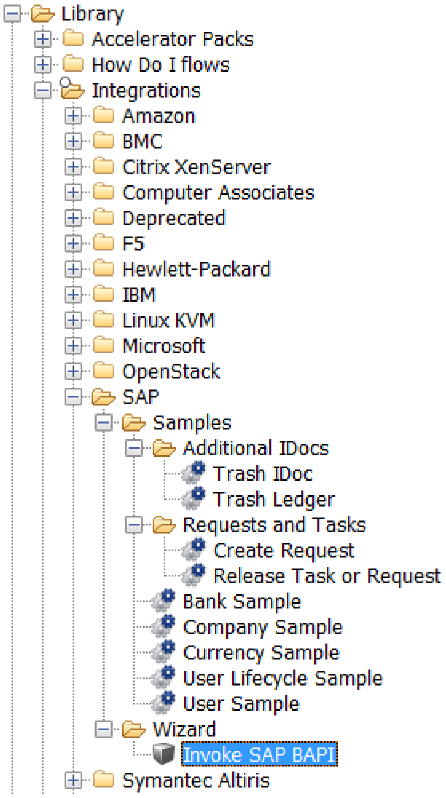
8. Restart the RAS service on all RASes in the remote OO deployment.

9. On Windows 2008 and Windows 2008 R2 servers, you must have administrative privileges to install OO-SAP Content. To get administrative privileges, follow these steps:

   a. From the **Start** menu select **All Programs**, and then **Accessories**.

   b. Right-click **Command Prompt** and then select **Run** as the administrator option.

After the installation completes, the SAP content is available in Studio.



Note:

Although the installation process finishes successfully, the following information is added to the RAS **wrapper.log** file, located in the **operations-orchestration-install-path\RAS\Java\Default\webapp\logs** folder.

```
WARN WrapperSimpleAppMain (14:00:34,525) jes - in getActions()
```

```
while inspecting com/hp/content/actions/SAP/
SAPDestinationDataProvider.
classjava.lang.NoClassDefFoundError:com/sap/conn
/jco/ext
/DestinationDataProvider
```

This is due to the dependency on the SAP Java Connector, which must be resolved before using the operation.

SAP Connector Installation Instructions

After installing the SAP content, the Java Connector (JCo) must be added before attempting to execute the operation. The installation instructions are also provided in the generic operation's description tab, under the **Notes** section.

Prior to using this operation, follow the steps below:

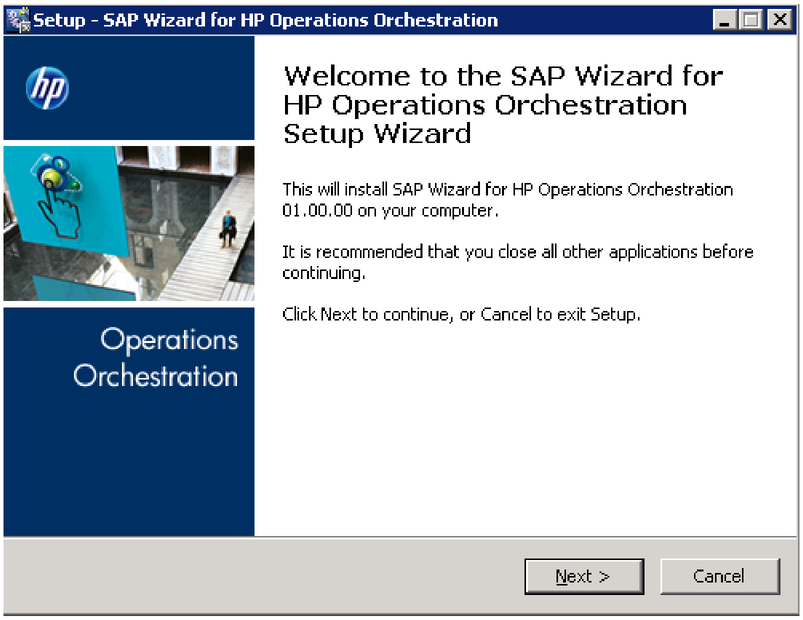
1. Download the SAP Java Connector (JCo) from the SAP Service Marketplace at http://service.sap.com/connectors where you can find all available distribution packages for the various supported platforms and processors.
2. Install JCo by unzipping the appropriate distribution package into an arbitrary directory{ **sapjco3-install-path}**:
   a. For Windows operating systems, copy **sapjco3.jar** and **sapjco3.dll** to the **operations-orchestration-install-path\RAS\Java\Default\repository\lib\SAP** folder. The SAP folder does not exist and you need to manually create it in advance.

   b. For Linux operating systems, copy **sapjco3.jar** and **libsapjco3.so** to the **operations-orchestration-install-path\RAS\Java\Default\repository\lib\SAP** folder. The SAP folder does not exist and you need to manually create it in advance.
3. Add {_sapjco3-install-path} to the PATH environment variable.
4. Restart the RSJRAS service.

SAP Wizard Installation Instructions

1. Double-click the **SAP Wizard for HP Operations Orchestration installation** icon.

   The Setup Welcome Page appears.



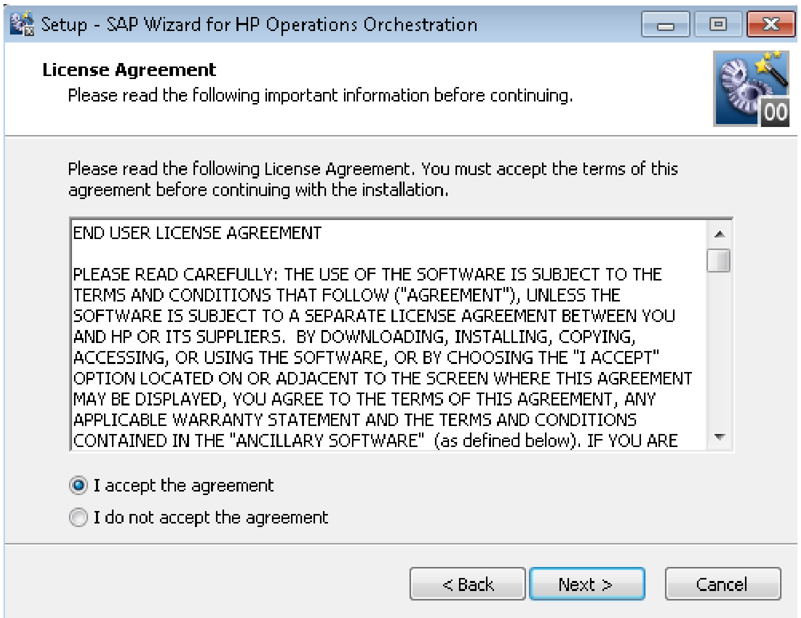
Welcome to the SAP Wizard for HP Operations Orchestration Setup Wizard

This will install SAP Wizard for HP Operations Orchestration 01.00.00 on your computer.

It is recommended that you close all other applications before continuing.

Click Next to continue, or Cancel to exit Setup.

2. Click **Next**. to open the Licence Agreement page.



License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

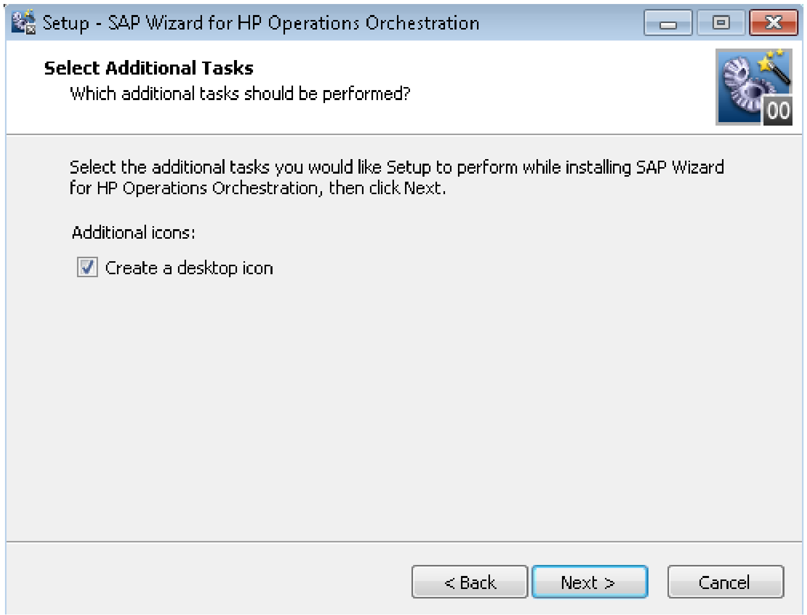
END USER LICENSE AGREEMENT

PLEASE READ CAREFULLY: THE USE OF THE SOFTWARE IS SUBJECT TO THE TERMS AND CONDITIONS THAT FOLLOW ("AGREEMENT"), UNLESS THE SOFTWARE IS SUBJECT TO A SEPARATE LICENSE AGREEMENT BETWEEN YOU AND HP OR ITS SUPPLIERS. BY DOWNLOADING, INSTALLING, COPYING, ACCESSING, OR USING THE SOFTWARE, OR BY CHOOSING THE "I ACCEPT" OPTION LOCATED ON OR ADJACENT TO THE SCREEN WHERE THIS AGREEMENT MAY BE DISPLAYED, YOU AGREE TO THE TERMS OF THIS AGREEMENT, ANY APPLICABLE WARRANTY STATEMENT AND THE TERMS AND CONDITIONS CONTAINED IN THE "ANCILLARY SOFTWARE" (as defined below). IF YOU ARE

● I accept the agreement

○ I do not accept the agreement

3. Accept the licence agreement and click **Next**.



Setup - SAP Wizard for HP Operations Orchestration

Select Additional Tasks

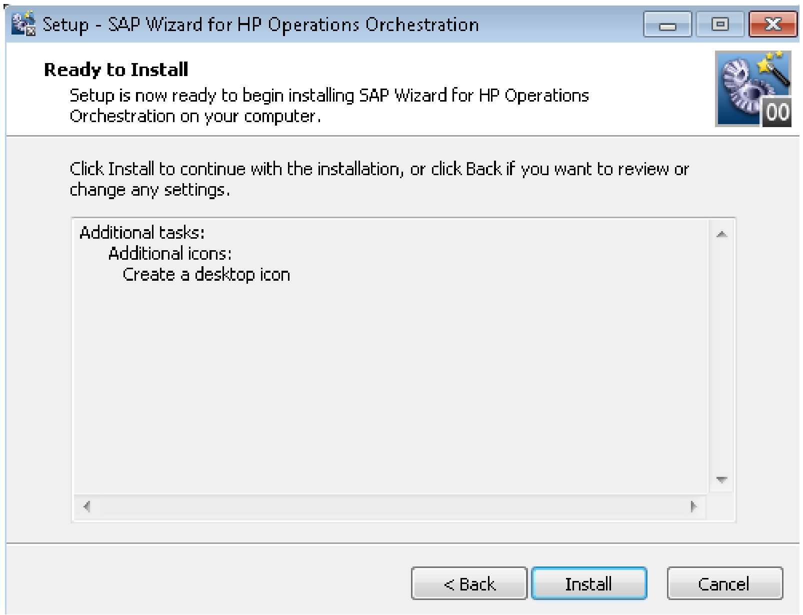
Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing SAP Wizard for HP Operations Orchestration, then click Next.

Additional icons:

☑ Create a desktop icon

4. Choose whether to create a desktop icon and click **Next**.



Setup - SAP Wizard for HP Operations Orchestration

Ready to Install

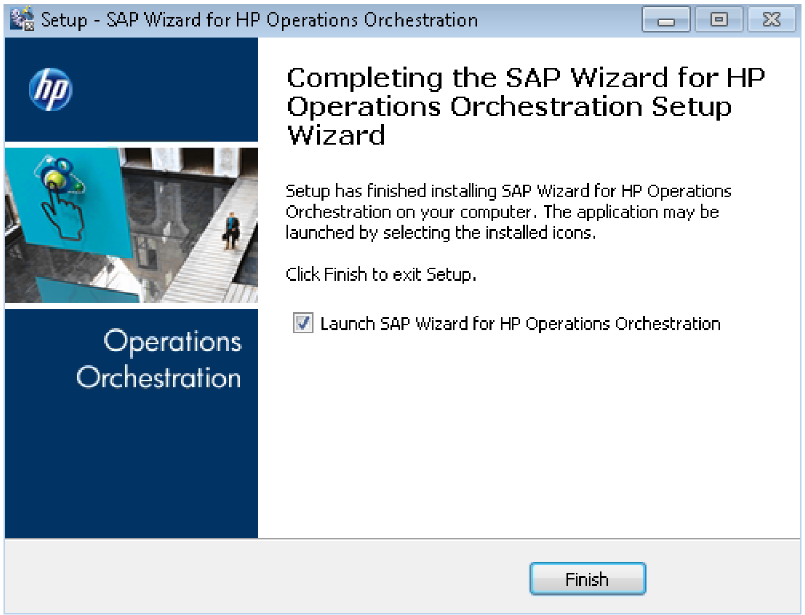
Setup is now ready to begin installing SAP Wizard for HP Operations Orchestration on your computer.

Click Install to continue with the installation, or click Back if you want to review or change any settings.

```
Additional tasks:
    Additional icons:
      Create a desktop icon
```

5. Click **Next** if the installation if the previous installation settings are correct.

6. If the installation process completed successfully, the completion page opens.



Completing the SAP Wizard for HP Operations Orchestration Setup Wizard

Setup has finished installing SAP Wizard for HP Operations Orchestration on your computer. The application may be launched by selecting the installed icons.

Click Finish to exit Setup.

☑ Launch SAP Wizard for HP Operations Orchestration

Click **Finish** to complete the installation.

Note: When the installation is finished, the two files for the SAP Java Connector (**sapjco3.jar** and **sapjco3.dll**) must be copied to the **operations-orchestration-install-path\Studio\tools\sapwizard\lib** folder.

Do not launch the wizard until these files are placed in this location. Otherwise the wizard's execution fails.

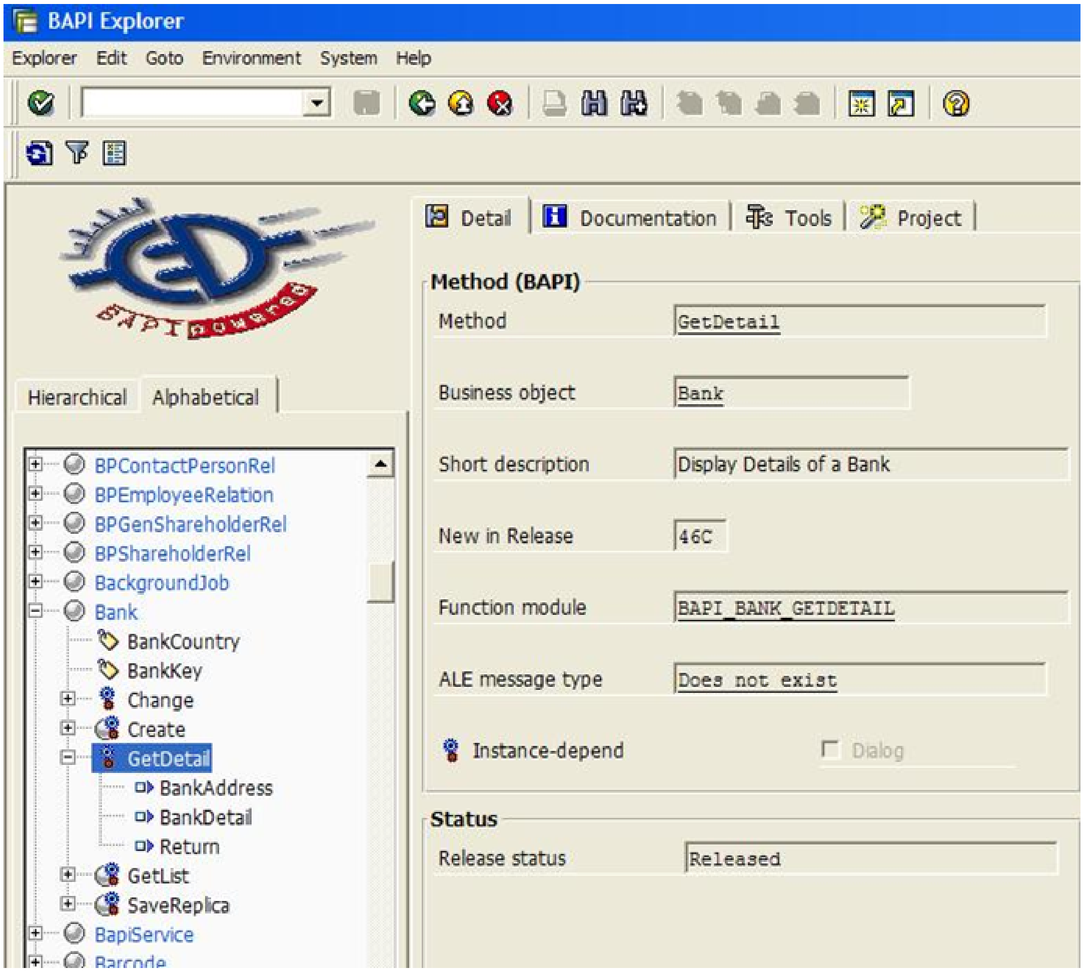
Chapter 4

Example of Using the Generic Invoke SAP BAPI Operation

In this example, you use the generic **Invoke SAP BAPI** operation to display the details of a bank by executing the GetDetail method of the Bank Business object. For this, you must uniquely identify the bank from which to list the details by following these tasks:

Step 1 — Locate the BAPI in the SAP GUI

1. Open BAPI Explorer as described in "Access the Available BAPIs in the SAP System" on page 10 and locate the Bank object.
2. Expand the **Bank** tree item and then select the **GetDetail** method to obtain information about the method you want to call.



Method (BAPI)

Method: GetDetail

Business object: Bank

Short description: Display Details of a Bank

New in Release: 46C

Function module: BAPI_BANK_GETDETAIL

ALE message type: Does not exist

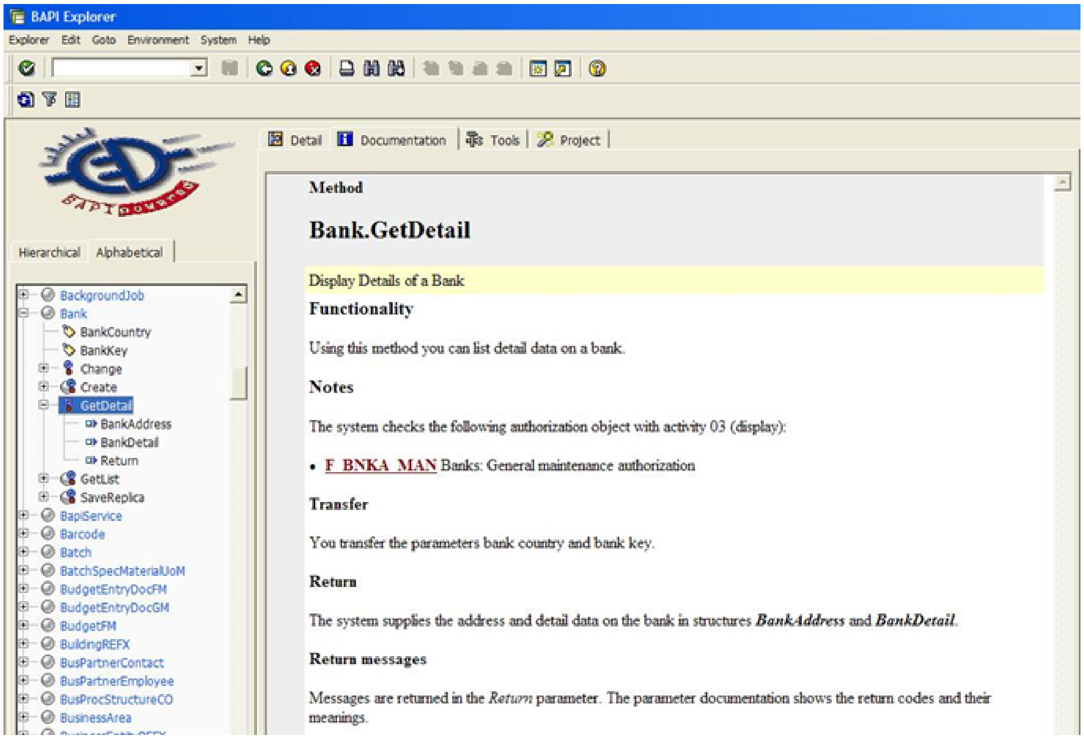
Instance-depend □ Dialog

Status

Release status: Released

The **Function** module associated to this method is named **BAPI_BANK_GETDETAIL**. This is the value that must be passed to the **bapi** input of the **Invoke SAP BAPI** operation in this example. Note that the general naming convention for **Function** modules is **BAPI_ ObjectName_MethodName**, where **ObjectName** and **MethodName** are written in all capital letters.

3. Click the **Documentation** tab to read a description of the method.



Method

Bank.GetDetail

Display Details of a Bank

Functionality

Using this method you can list detail data on a bank.

Notes

The system checks the following authorization object with activity 03 (display):

- F_BNKA_MAN Banks: General maintenance authorization

Transfer

You transfer the parameters bank country and bank key.

Return

The system supplies the address and detail data on the bank in structures ***BankAddress*** and ***BankDetail***.

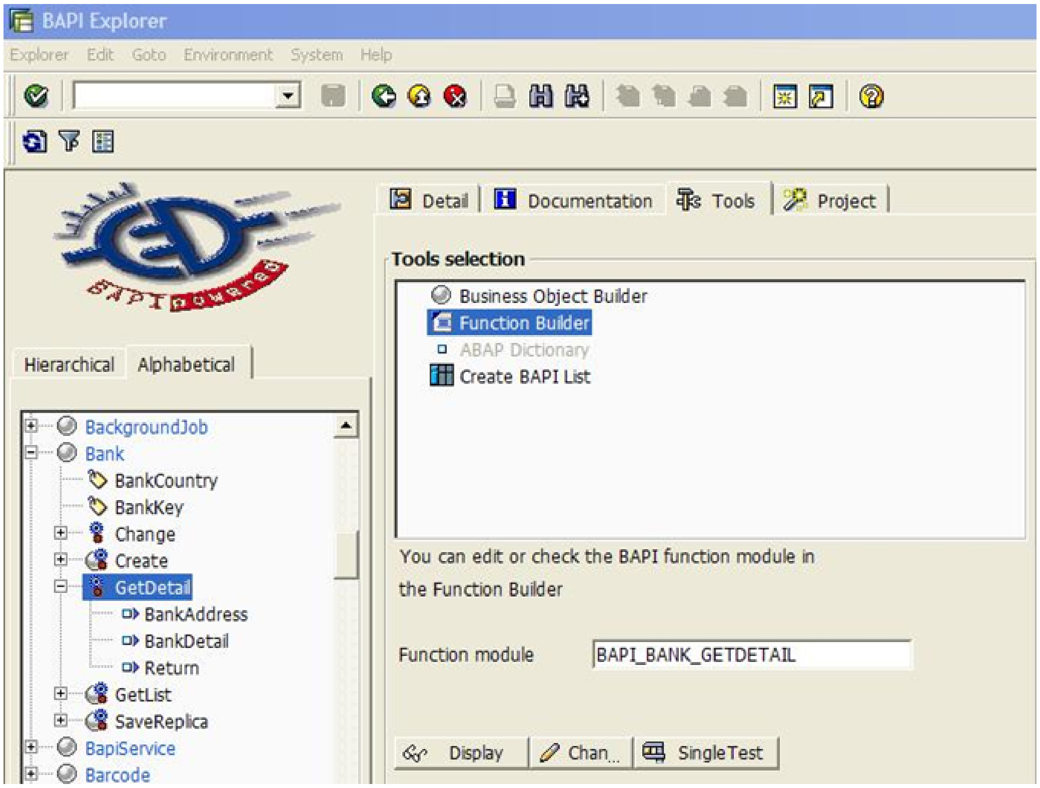
Return messages

Messages are returned in the *Return* parameter. The parameter documentation shows the return codes and their meanings.

4. To identify the specific import, export, and table parameters of the BAPI:
   a. Click the **Tools** tab.

   b. Select **Function Builder** and then click the **Display** button.

   c. Choose the appropriate tab for a list of specific parameters.

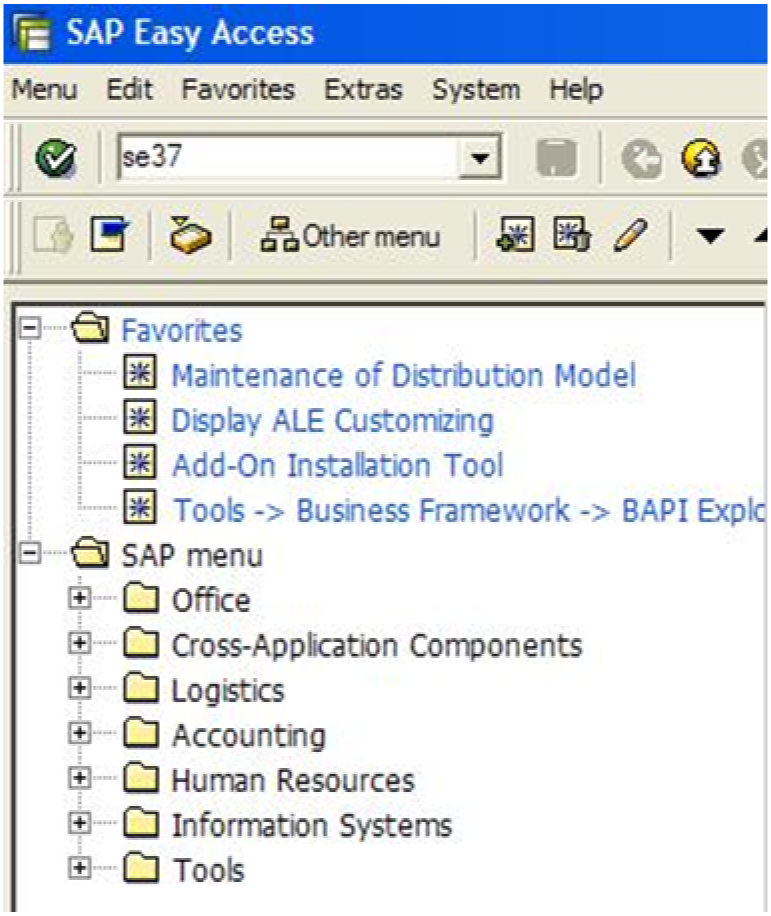


You can edit or check the BAPI function module in the Function Builder

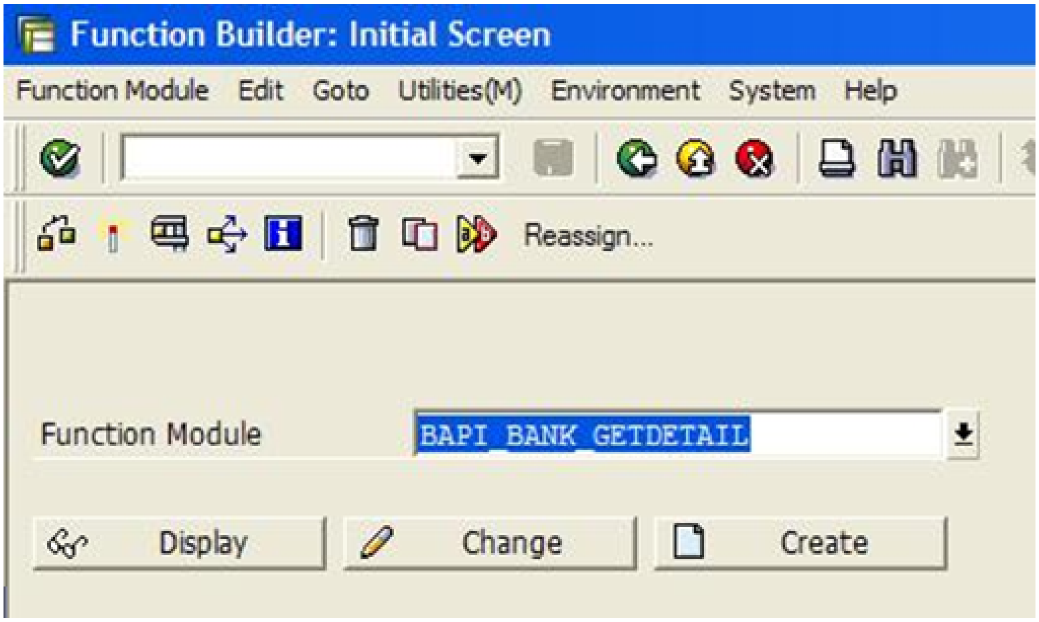
Function module: BAPI_BANK_GETDETAIL

Or

a. Open the Function Builder by entering the **se37** transaction code (as shown below).



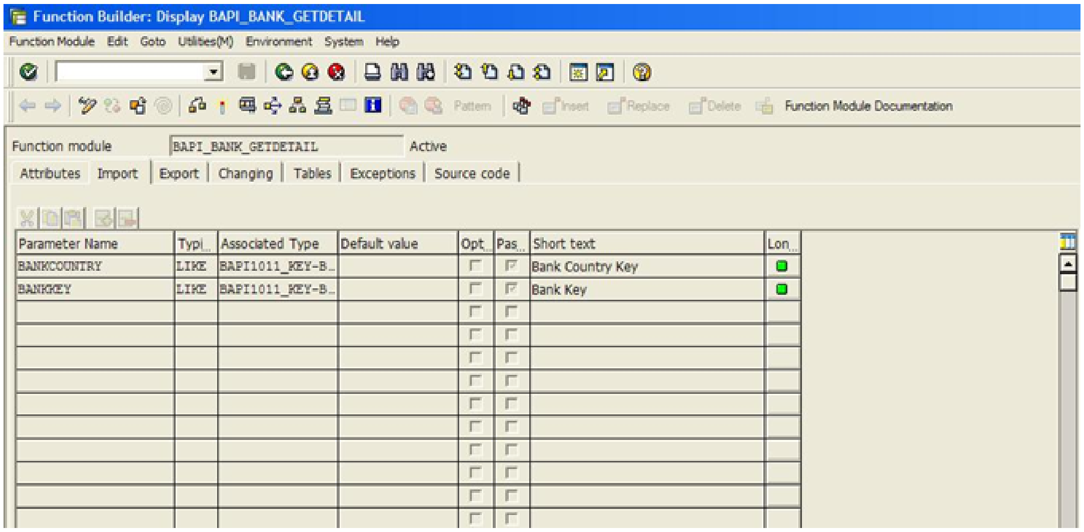
b. Type the name of the Function Module and then click the **Display** button.


Function Builder: Initial Screen
Function Module Edit Goto Utilities(M) Environment System Help
Reassign...
Display
Change
Create

Function Module BAPI_BANK_GETDETAIL

Step 2 — Add the BAPI Specific Inputs

1. To view the import parameters that can be used with the generic **Invoke SAP BAPI** operation to call the BAPI_BANK_GETDETAIL function, select the **Import** tab.


Function Builder: Display BAPI_BANK_GETDETAIL
Function Module Edit Goto Utilities(M) Environment System Help
Pattern
Insert
Replace
Delete
Function Module Documentation

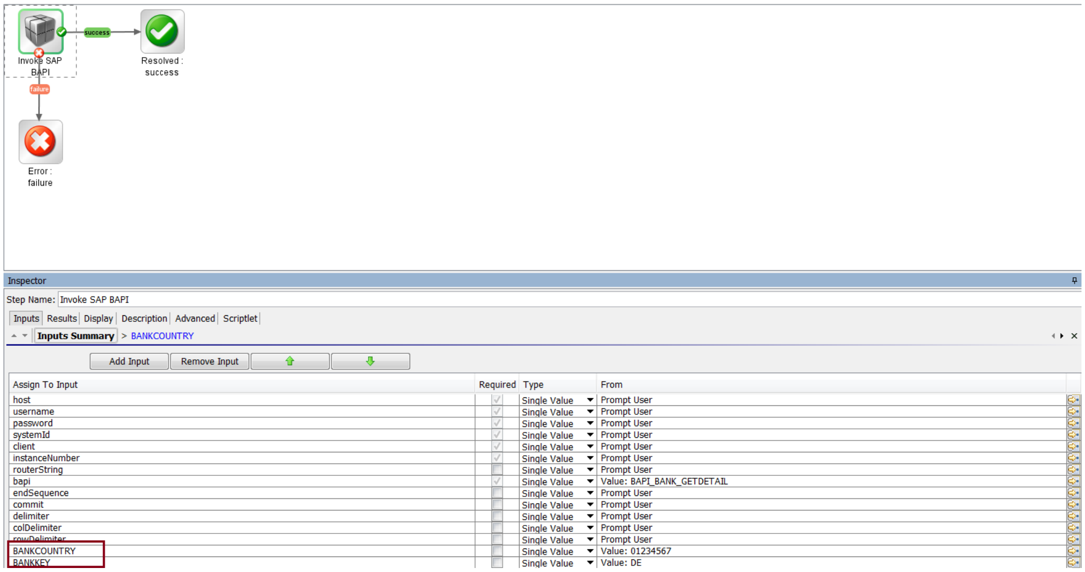
Function module BAPI_BANK_GETDETAIL Active

Attributes | Import | Export | Changing | Tables | Exceptions | Source code

Parameter Name	Typi..	Associated Type	Default value	Opt..	Pas..	Short text	Lon...
BANKCOUNTRY	LIKE	BAPI1011_KEY-B..		☐	☑	Bank Country Key	🟩
BANKKEY	LIKE	BAPI1011_KEY-B..		☐	☑	Bank Key	🟩

The BAPI_BANK_GETDETAIL function has two import parameters named BANKKEY and BANKCOUNTRY that allow you to identify the Bank object on which the method is called.

2. To pass the values for these parameters from OO, you must add BANKKEY and BANKCOUNTRY as inputs in the generic **Invoke SAP BAPI** operation.



Step Name: Invoke SAP BAPI

Inputs | Results | Display | Description | Advanced | Scriptlet

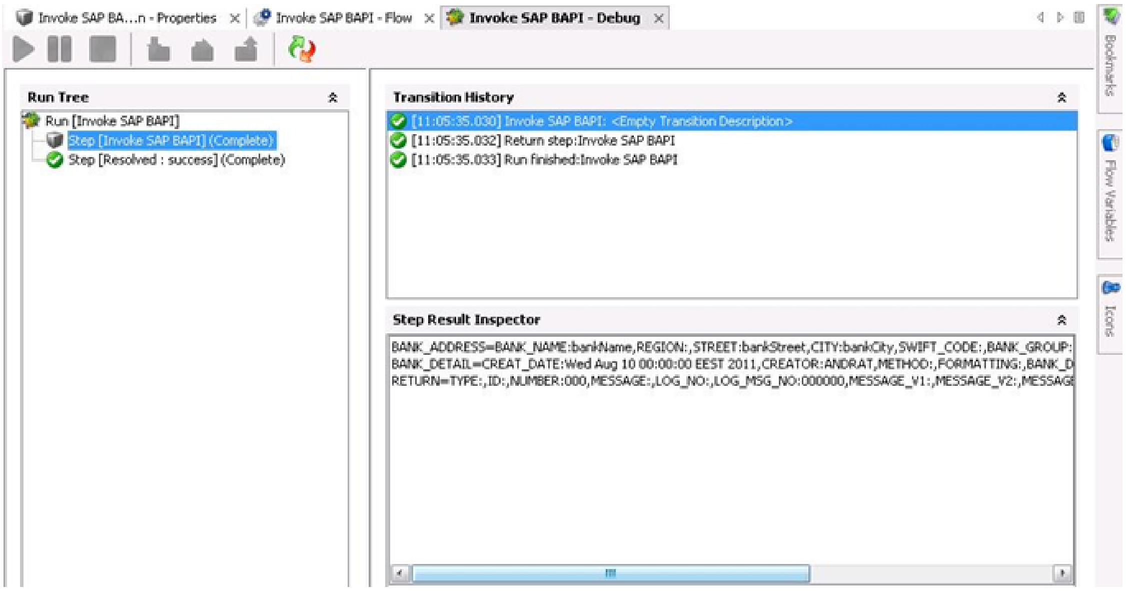
Inputs Summary > BANKCOUNTRY

Assign To Input	Required	Type	From
host	☑	Single Value	Prompt User
username	☑	Single Value	Prompt User
password	☑	Single Value	Prompt User
systemId	☑	Single Value	Prompt User
client	☑	Single Value	Prompt User
instanceNumber	☑	Single Value	Prompt User
routerString	☐	Single Value	Prompt User
bapi	☑	Single Value	Value: BAPI_BANK_GETDETAIL
endSequence	☐	Single Value	Prompt User
commit	☐	Single Value	Prompt User
delimiter	☐	Single Value	Prompt User
colDelimiter	☐	Single Value	Prompt User
rowDelimiter	☐	Single Value	Prompt User
BANKCOUNTRY	☐	Single Value	Value: 01234567
BANKKEY	☐	Single Value	Value: DE

3. Follow the same procedure for customizing the generic Invoke SAP BAPI operation to invoke any BAPI for which you wish to pass specific import parameters values.

Step 3 — Add the BAPI Specific Results

The **Invoke SAP BAPI** operation returns the table and export parameters specific to the called BAPI in the returnResult result. When you use the **Invoke SAP BAPI** operation, you can customize this output by applying filters to extract results for each of the desired SAP parameters. For example, when invoking BAPI_BANK_GETDETAIL, the returnResult would appear in the OO Studio Debug screen as shown below.



Run Tree

- Run [Invoke SAP BAPI]
  - Step [Invoke SAP BAPI] (Complete)
  - Step [Resolved : success] (Complete)

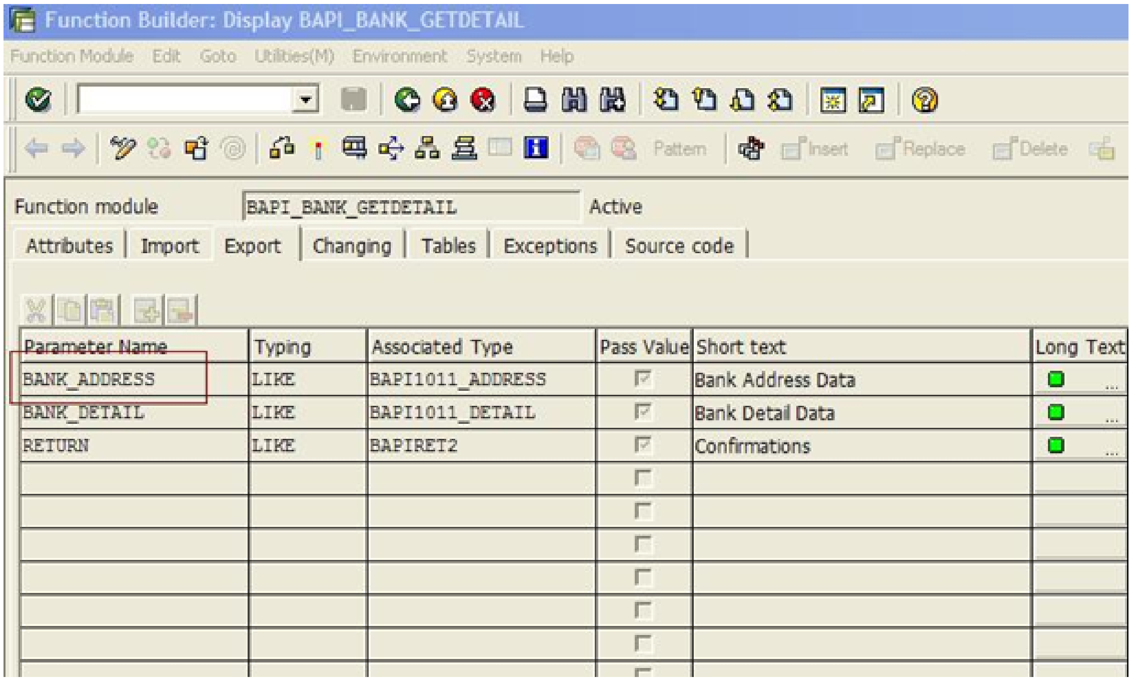
Transition History

- [11:05:35.030] Invoke SAP BAPI: <Empty Transition Description>
- [11:05:35.032] Return step:Invoke SAP BAPI
- [11:05:35.033] Run finished:Invoke SAP BAPI

Step Result Inspector

```
BANK_ADDRESS=BANK_NAME:bankName,REGION:,STREET:bankStreet,CITY:bankCity,SWIFT_CODE:,BANK_GROUP:
BANK_DETAIL=CREAT_DATE:Wed Aug 10 00:00:00 EEST 2011,CREATOR:ANDRAT,METHOD:,FORMATTING:,BANK_D
RETURN=TYPE:,ID:,NUMBER:000,MESSAGE:,LOG_NO:,LOG_MSG_NO:000000,MESSAGE_V1:,MESSAGE_V2:,MESSAGE
```

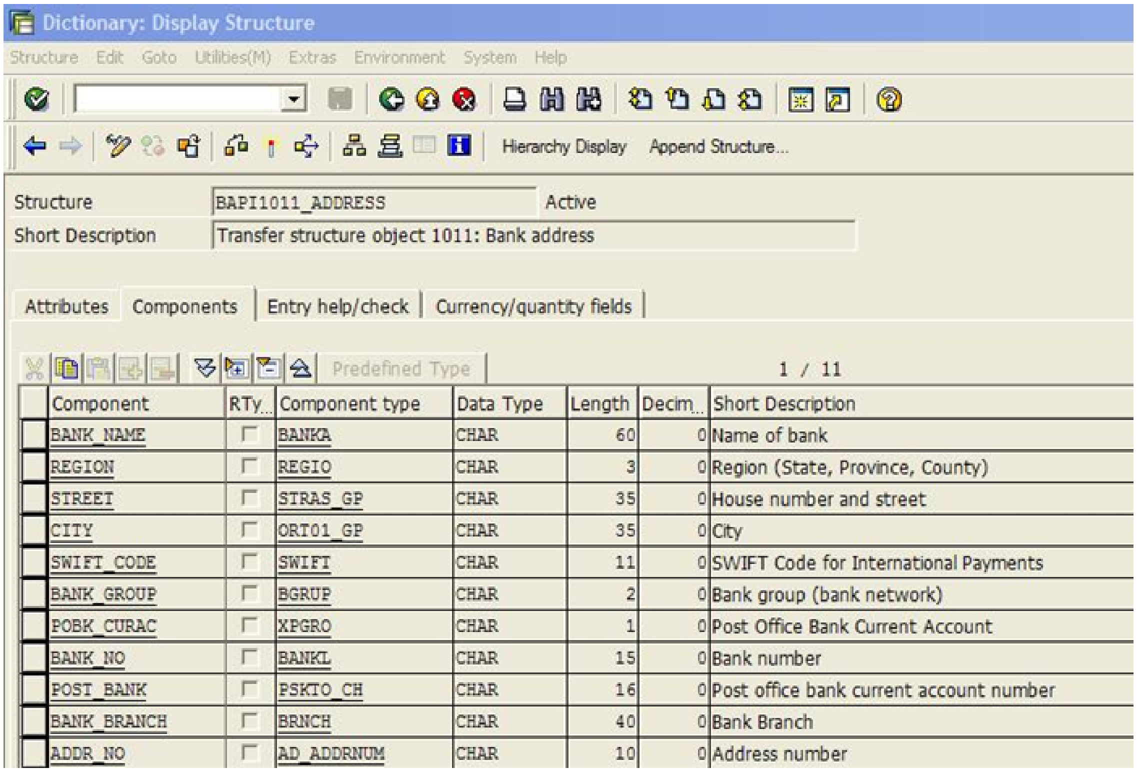
You can check for the table and export parameters related to the **BAPI_BANK_GETDETAIL** in SAP and see that the BAPI has an export parameter named BANK_ADDRESS.



Function module BAPI_BANK_GETDETAIL Active

Parameter Name	Typing	Associated Type	Pass Value	Short text	Long Text
BANK_ADDRESS	LIKE	BAPI1011_ADDRESS	☑	Bank Address Data	🟩 ...
BANK_DETAIL	LIKE	BAPI1011_DETAIL	☑	Bank Detail Data	🟩 ...
RETURN	LIKE	BAPIRET2	☑	Confirmations	🟩 ...
			☐		
			☐		
			☐		
			☐		
			☐		
			☐		

This is a structure parameter and its fields are enumerated if we open its **Associated Type**.

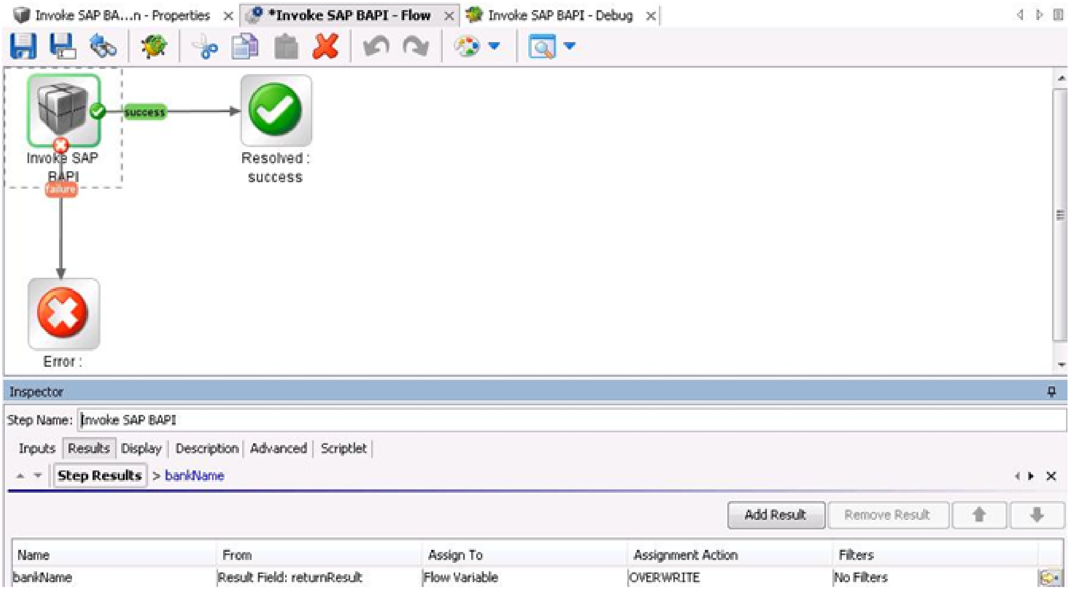


Structure BAPI1011_ADDRESS Active
Short Description Transfer structure object 1011: Bank address

Component	RTy...	Component type	Data Type	Length	Decim...	Short Description
BANK_NAME	□	BANKA	CHAR	60	0	Name of bank
REGION	□	REGIO	CHAR	3	0	Region (State, Province, County)
STREET	□	STRAS_GP	CHAR	35	0	House number and street
CITY	□	ORT01_GP	CHAR	35	0	City
SWIFT_CODE	□	SWIFT	CHAR	11	0	SWIFT Code for International Payments
BANK_GROUP	□	BGRUP	CHAR	2	0	Bank group (bank network)
POBK_CURAC	□	XPGRO	CHAR	1	0	Post Office Bank Current Account
BANK_NO	□	BANKL	CHAR	15	0	Bank number
POST_BANK	□	PSKTO_CH	CHAR	16	0	Post office bank current account number
BANK_BRANCH	□	BRNCH	CHAR	40	0	Bank Branch
ADDR_NO	□	AD_ADDRNUM	CHAR	10	0	Address number

To add the BAPI specific results:

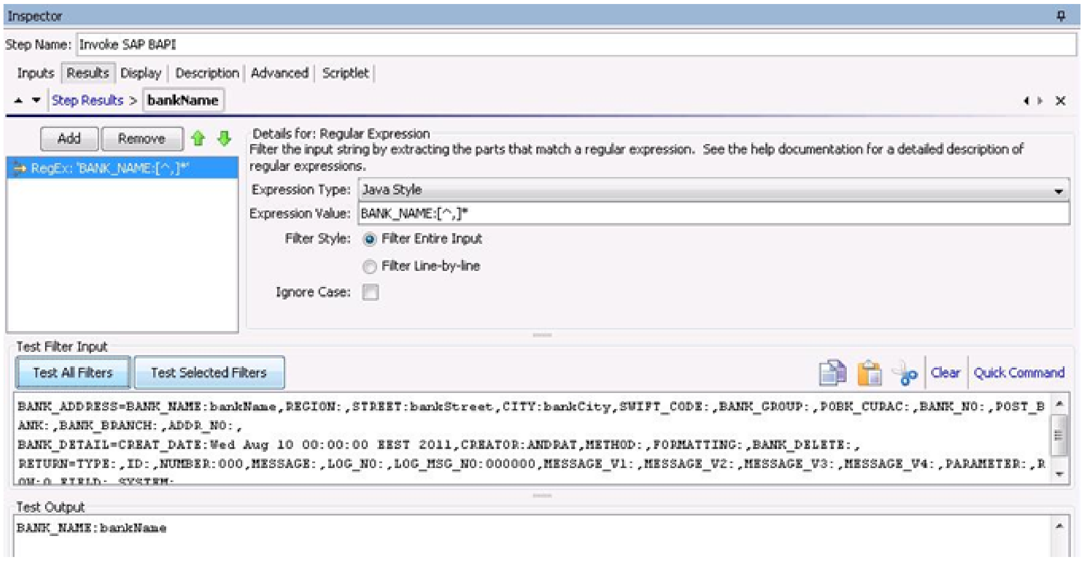
1. To obtain the **BANK_NAME** field of the BANK_ADDRESS parameter, filter the **Invoke SAP BAPI** operation's primary output. Create a step result named **bankName** which gets its value from the operation's **returnResult**.



Step Name: Invoke SAP BAPI

Name	From	Assign To	Assignment Action	Filters
bankName	Result Field: returnResult	Flow Variable	OVERWRITE	No Filters

2. Use a regular expression to extract the BANK_NAME from the **returnResult**.



Step Name: Invoke SAP BAPI

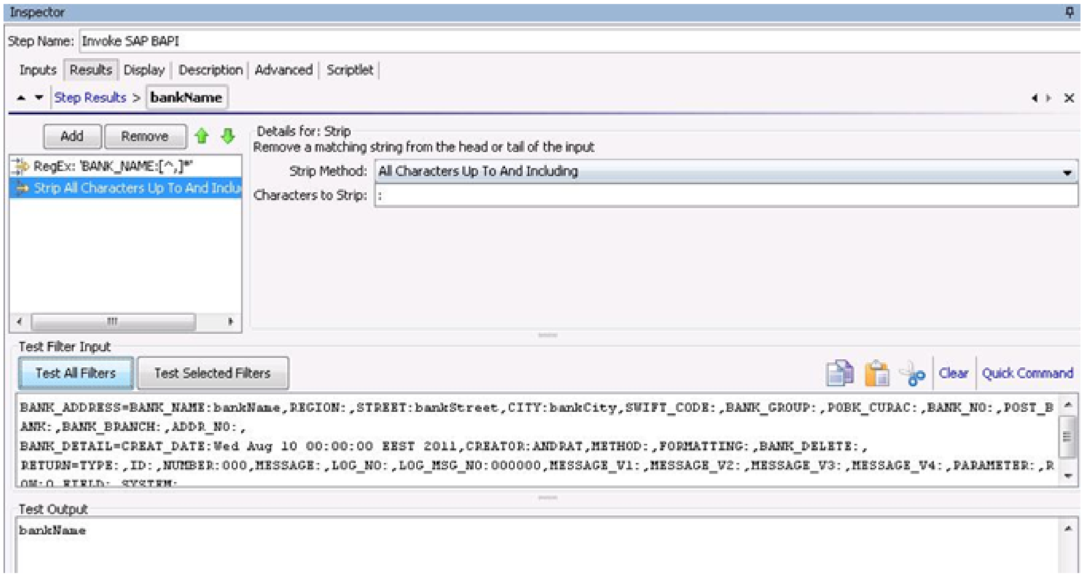
Details for: Regular Expression
Filter the input string by extracting the parts that match a regular expression. See the help documentation for a detailed description of regular expressions.

Expression Type: Java Style
Expression Value: BANK_NAME:[^,]*
Filter Style: ● Filter Entire Input ○ Filter Line-by-line
Ignore Case: □

```
BANK_ADDRESS=BANK_NAME:bankName,REGION:,STREET:bankStreet,CITY:bankCity,SWIFT_CODE:,BANK_GROUP:,POBK_CURAC:,BANK_NO:,POST_B
ANK:,BANK_BRANCH:,ADDR_NO:,
BANK_DETAIL=CREAT_DATE:Wed Aug 10 00:00:00 EEST 2011,CREATOR:ANDRAT,METHOD:,FORMATTING:,BANK_DELETE:,
RETURN=TYPE:,ID:,NUMBER:000,MESSAGE:,LOG_NO:,LOG_MSG_NO:000000,MESSAGE_V1:,MESSAGE_V2:,MESSAGE_V3:,MESSAGE_V4:,PARAMETER:,R
OW:0,FIELD:,SYSTEM:
```

```
BANK_NAME:bankName
```

3. If you want to eliminate the name of the SAP field, you can use a **Strip** filter to remove BANK_ NAME from the result.



Step Name: Invoke SAP BAPI

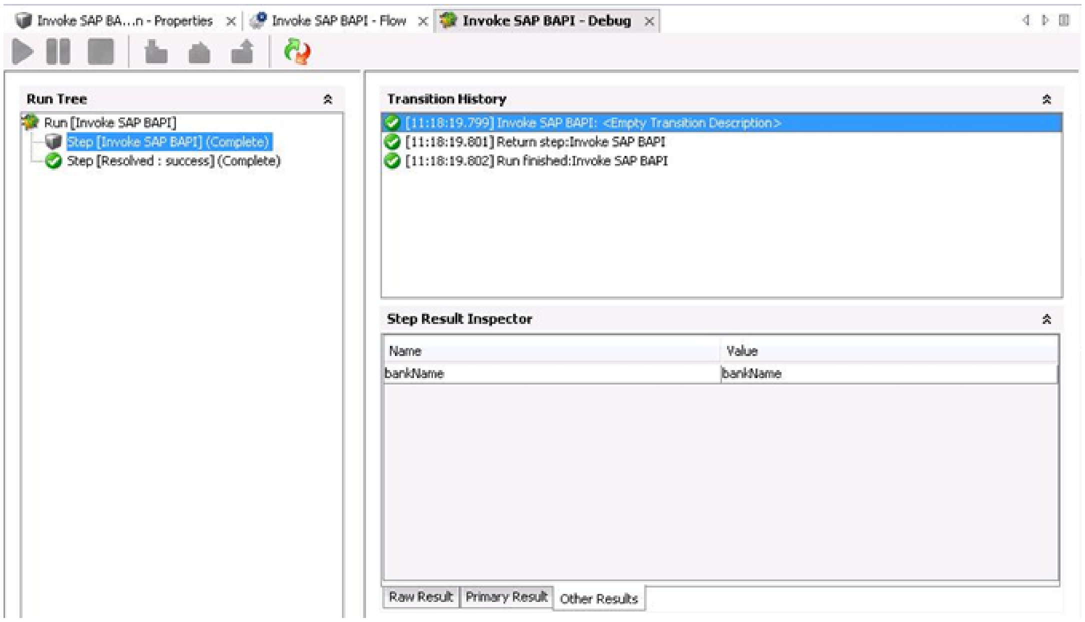
Details for: Strip
Remove a matching string from the head or tail of the input

Strip Method: All Characters Up To And Including
Characters to Strip: :

```
BANK_ADDRESS=BANK_NAME:bankName,REGION:,STREET:bankStreet,CITY:bankCity,SWIFT_CODE:,BANK_GROUP:,POBK_CURAC:,BANK_NO:,POST_B
ANK:,BANK_BRANCH:,ADDR_NO:,
BANK_DETAIL=CREAT_DATE:Wed Aug 10 00:00:00 EEST 2011,CREATOR:ANDRAT,METHOD:,FORMATTING:,BANK_DELETE:,
RETURN=TYPE:,ID:,NUMBER:000,MESSAGE:,LOG_NO:,LOG_MSG_NO:000000,MESSAGE_V1:,MESSAGE_V2:,MESSAGE_V3:,MESSAGE_V4:,PARAMETER:,R
OW:0,FIELD:,SYSTEM:
```

```
bankName
```

When you execute this step again, **bankName** is obtained as a separate result.



```
[11:18:19.799] Invoke SAP BAPI: <Empty Transition Description>
[11:18:19.801] Return step:Invoke SAP BAPI
[11:18:19.802] Run finished:Invoke SAP BAPI
```

Name	Value
bankName	bankName

Chapter 5

Creating SAP Integration Flows

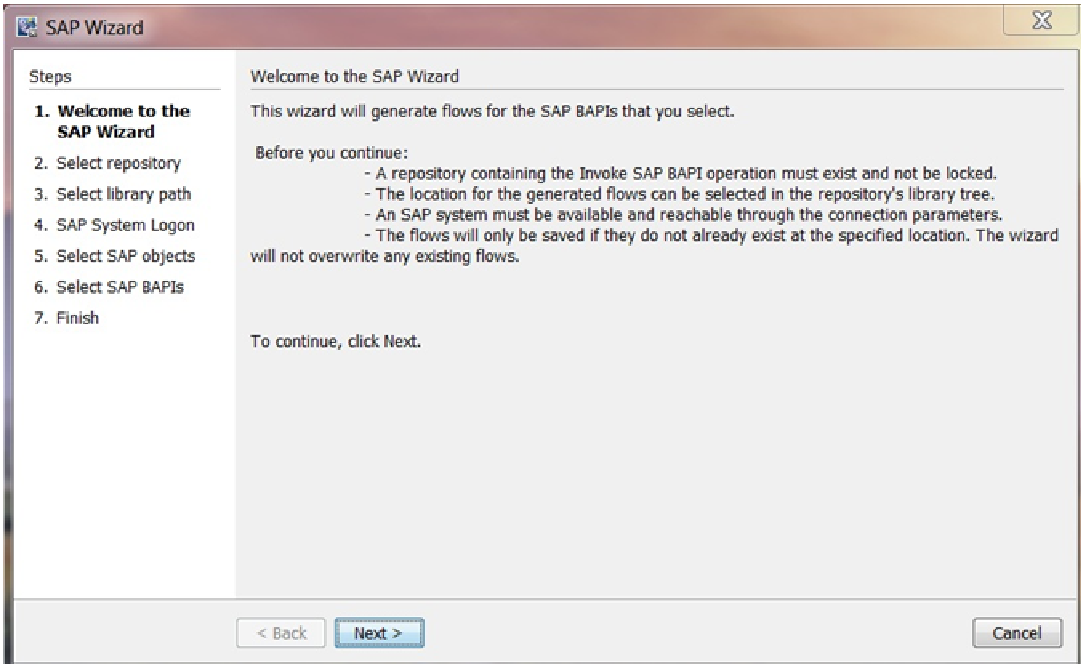
Using the procedures in "Example of Using the Generic Invoke SAP BAPI Operation" on page 23 to manually configure the inputs and results of the generic **Invoke SAP BAPI** operation to call a particular BAPI can be a complex process, especially when the BAPI has a large number of parameters associated. You can simplify this process by using the SAP Wizard to generate specialized flows that are already customized for the BAPIs you wish to invoke.

Using the SAP Wizard to Create OO Flows for Selected BAPIs

The SAP Wizard creates OO flows based on the BAPIs available in the SAP system, which you specify when you run the wizard. This tool is available in the OO home directory, under the **Wizards** folder.

To use the SAP Wizard to create an OO flow for a BAPI:

1. Start the SAP Wizard to open the Welcome page.



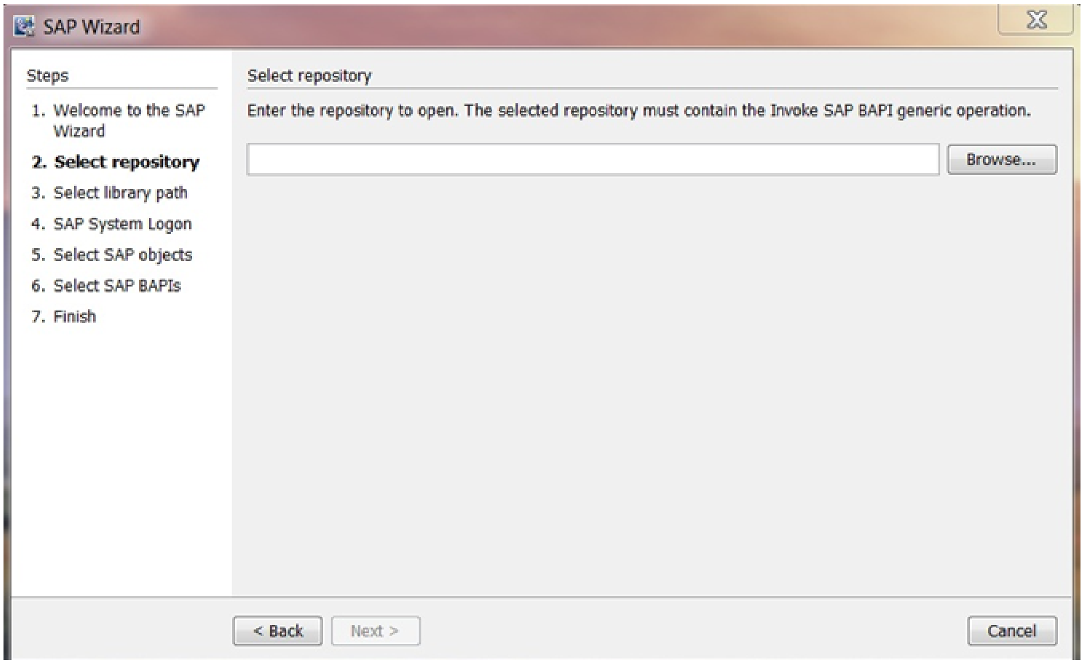
Welcome to the SAP Wizard

This wizard will generate flows for the SAP BAPIs that you select.

Before you continue:
- A repository containing the Invoke SAP BAPI operation must exist and not be locked.
- The location for the generated flows can be selected in the repository's library tree.
- An SAP system must be available and reachable through the connection parameters.
- The flows will only be saved if they do not already exist at the specified location. The wizard will not overwrite any existing flows.

To continue, click Next.

2. Click **Next** to open the Select Repository page.



SAP Wizard

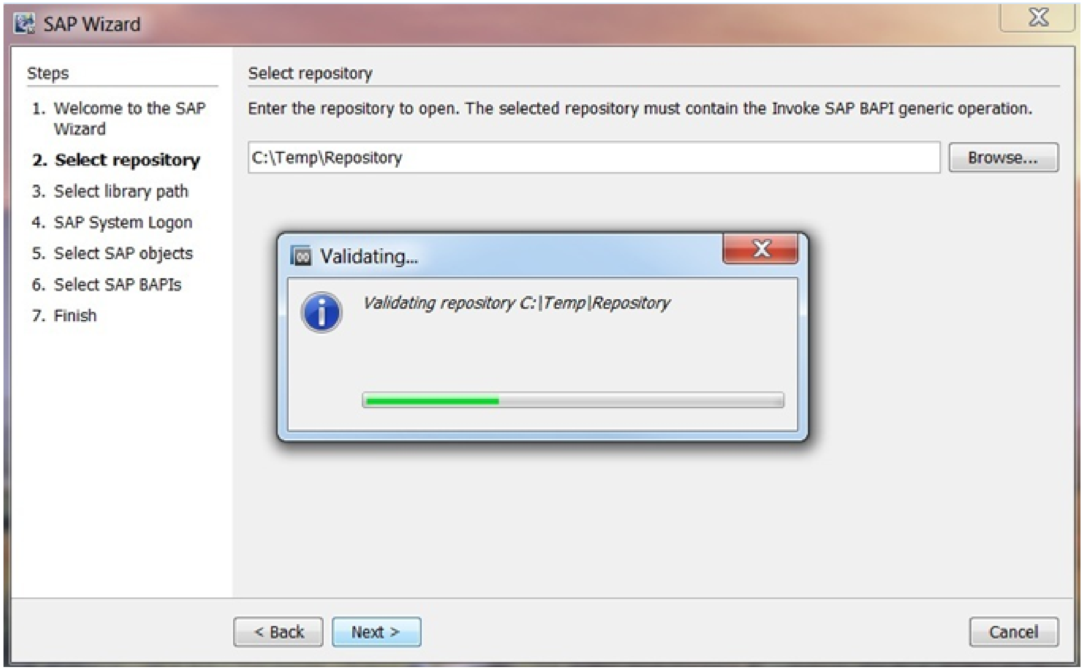
Steps

1. Welcome to the SAP Wizard
2. **Select repository**
3. Select library path
4. SAP System Logon
5. Select SAP objects
6. Select SAP BAPIs
7. Finish

Select repository

Enter the repository to open. The selected repository must contain the Invoke SAP BAPI generic operation.

3. Enter or select a repository for the flows you want to create, and then click **Next** .The SAP Wizard validates the repository.



SAP Wizard

Steps

1. Welcome to the SAP Wizard
2. **Select repository**
3. Select library path
4. SAP System Logon
5. Select SAP objects
6. Select SAP BAPIs
7. Finish

Select repository

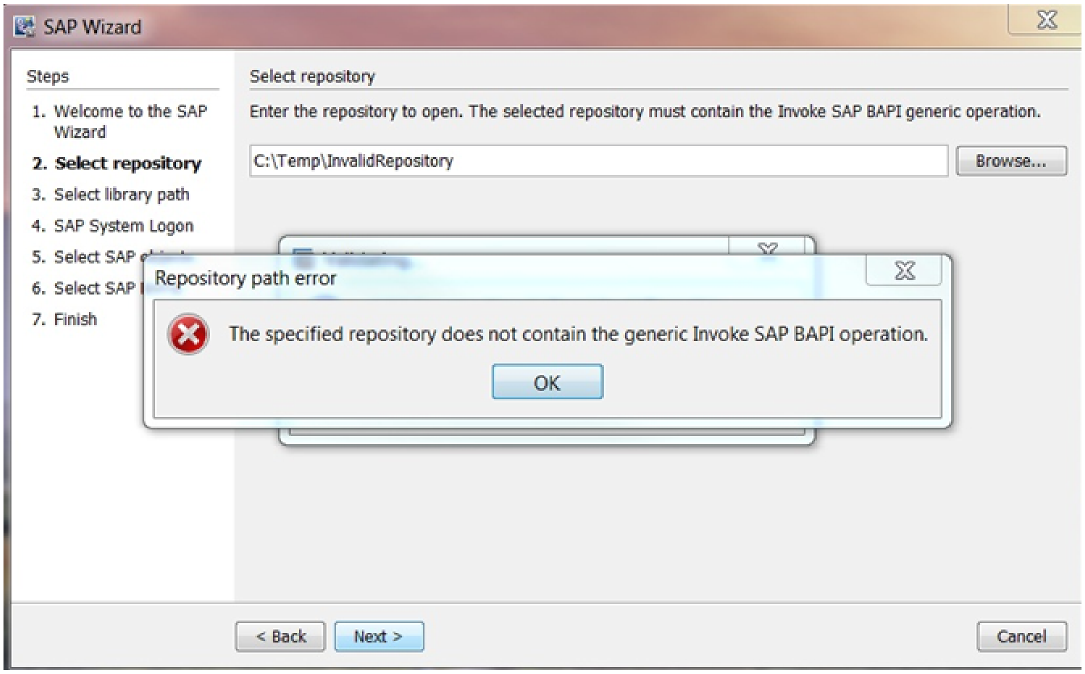
Enter the repository to open. The selected repository must contain the Invoke SAP BAPI generic operation.

C:\Temp\Repository

Validating...

Validating repository C:|Temp|Repository

Note: The specified repository must contain the generic **Invoke SAP BAPI** operation, otherwise it is considered invalid and this step fails. In addition, make sure that the repository is not locked (that is, it should not be opened in OO Studio).



Select repository

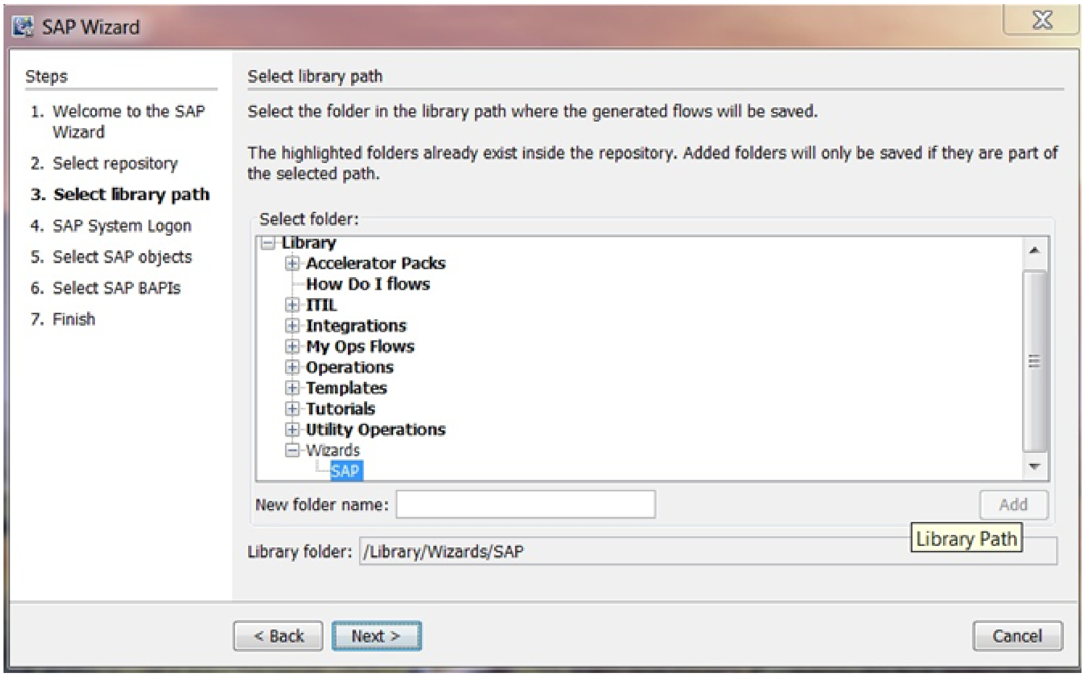
Enter the repository to open. The selected repository must contain the Invoke SAP BAPI generic operation.

C:\Temp\InvalidRepository

Repository path error

The specified repository does not contain the generic Invoke SAP BAPI operation.

If the specified repository is valid, the library path selection page appears.



Select library path

Select the folder in the library path where the generated flows will be saved.

The highlighted folders already exist inside the repository. Added folders will only be saved if they are part of the selected path.

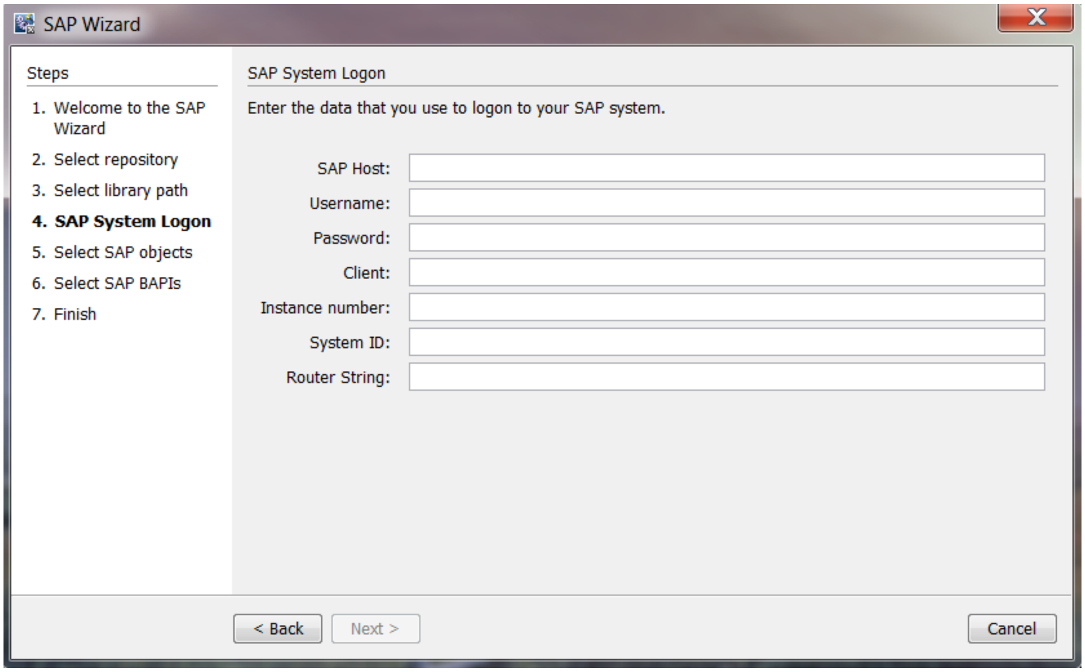
Select folder:

- Library
  - Accelerator Packs
  - How Do I flows
  - ITIL
  - Integrations
  - My Ops Flows
  - Operations
  - Templates
  - Tutorials
  - Utility Operations
  - Wizards
    - SAP

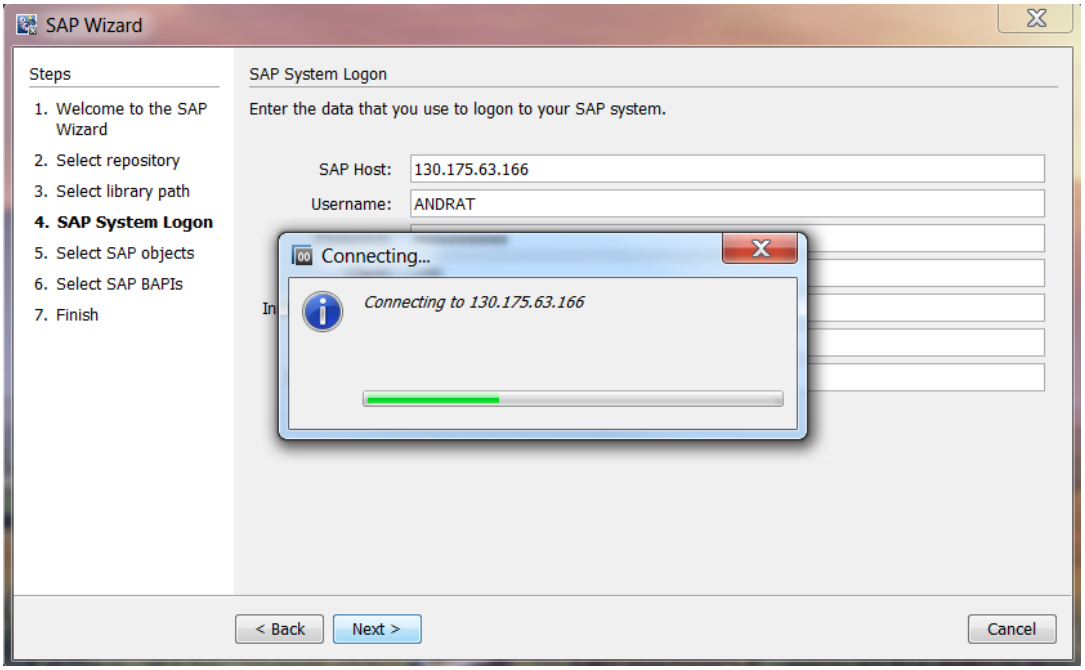
New folder name:

Library folder: /Library/Wizards/SAP

After selecting the library location for the generated flows, the logon page opens.



4. Enter the inputs and wait for the connection to be established.



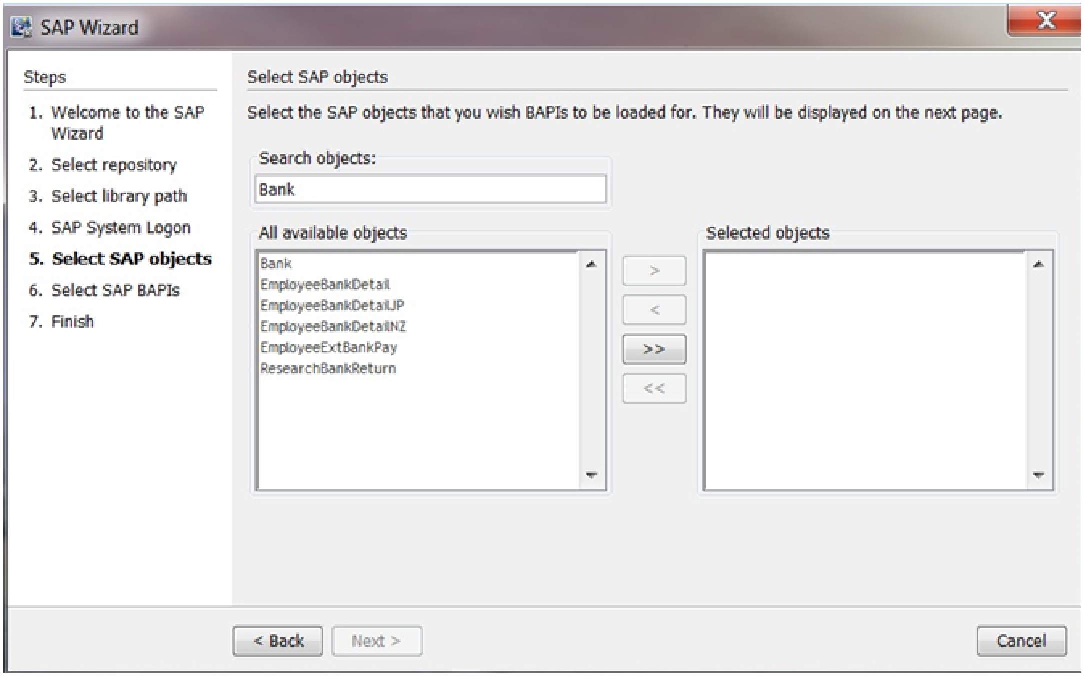
SAP Host: 130.175.63.166

Username: ANDRAT

Connecting to 130.175.63.166

If the connection succeeds, the next page lists all the Business Objects in the specified SAP system.

> **Note:** All available objects are included in the available objects list, although some may not have associated BAPIs. The search mechanism enables you to filter the list of objects based on the search term.



SAP Wizard

Steps

1. Welcome to the SAP Wizard
2. Select repository
3. Select library path
4. SAP System Logon
5. **Select SAP objects**
6. Select SAP BAPIs
7. Finish

Select SAP objects

Select the SAP objects that you wish BAPIs to be loaded for. They will be displayed on the next page.

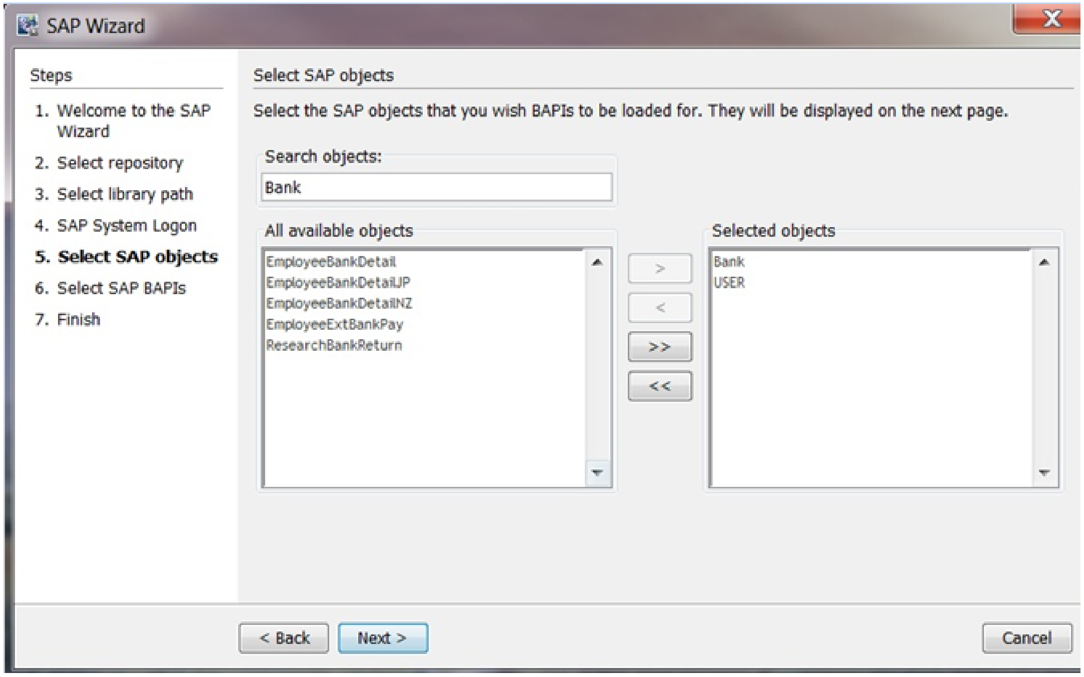
Search objects: Bank

All available objects:

Bank
EmployeeBankDetail
EmployeeBankDetailJP
EmployeeBankDetailNZ
EmployeeExtBankPay
ResearchBankReturn

Selected objects:

5. Select the Business Objects that have the BAPIs for which you want to create flows.



SAP Wizard

Steps

1. Welcome to the SAP Wizard
2. Select repository
3. Select library path
4. SAP System Logon
5. **Select SAP objects**
6. Select SAP BAPIs
7. Finish

Select SAP objects

Select the SAP objects that you wish BAPIs to be loaded for. They will be displayed on the next page.

Search objects: Bank

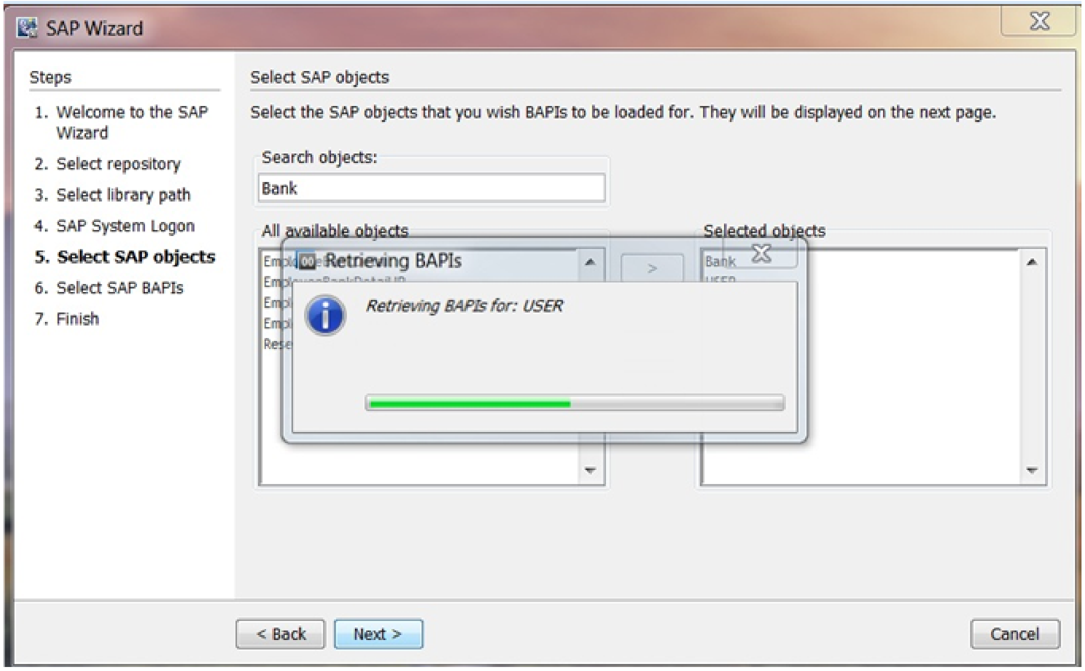
All available objects:

EmployeeBankDetail
EmployeeBankDetailJP
EmployeeBankDetailNZ
EmployeeExtBankPay
ResearchBankReturn

Selected objects:

Bank
USER

6. Click **Next**. The BAPIs related to the selected Business Object are retrieved.



Select SAP objects

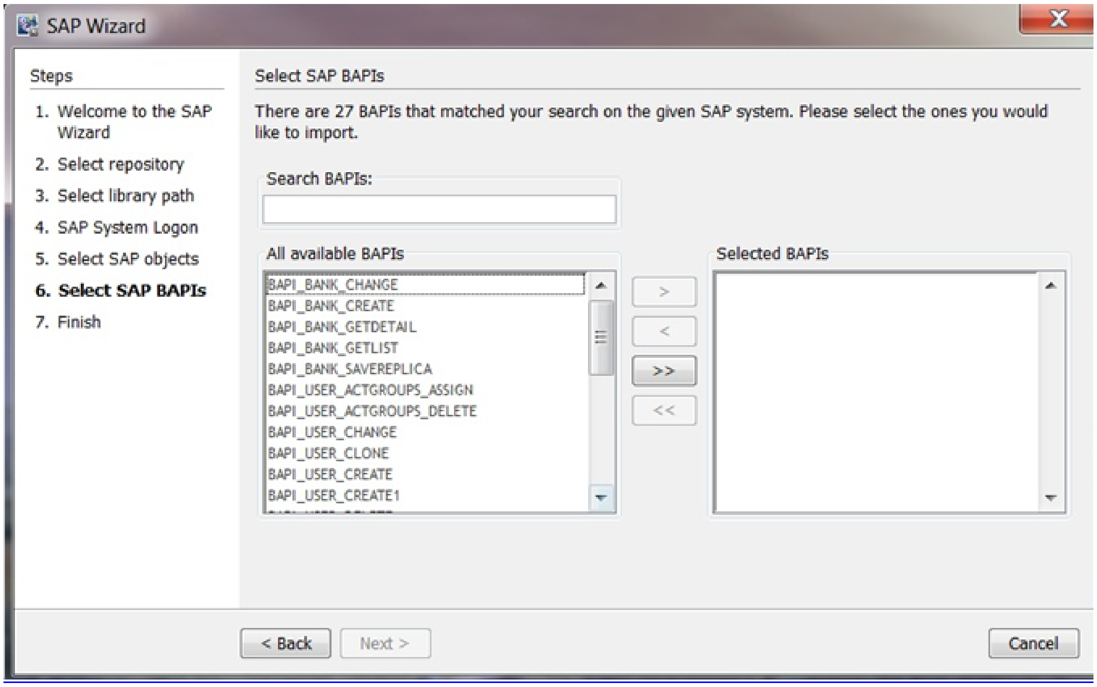
Select the SAP objects that you wish BAPIs to be loaded for. They will be displayed on the next page.

Search objects: Bank

Selected objects: Bank, USER

Retrieving BAPIs for: USER

After the required BAPIs have been retrieved, the SAP Wizard lists all the BAPIs of the selected Business Objects.



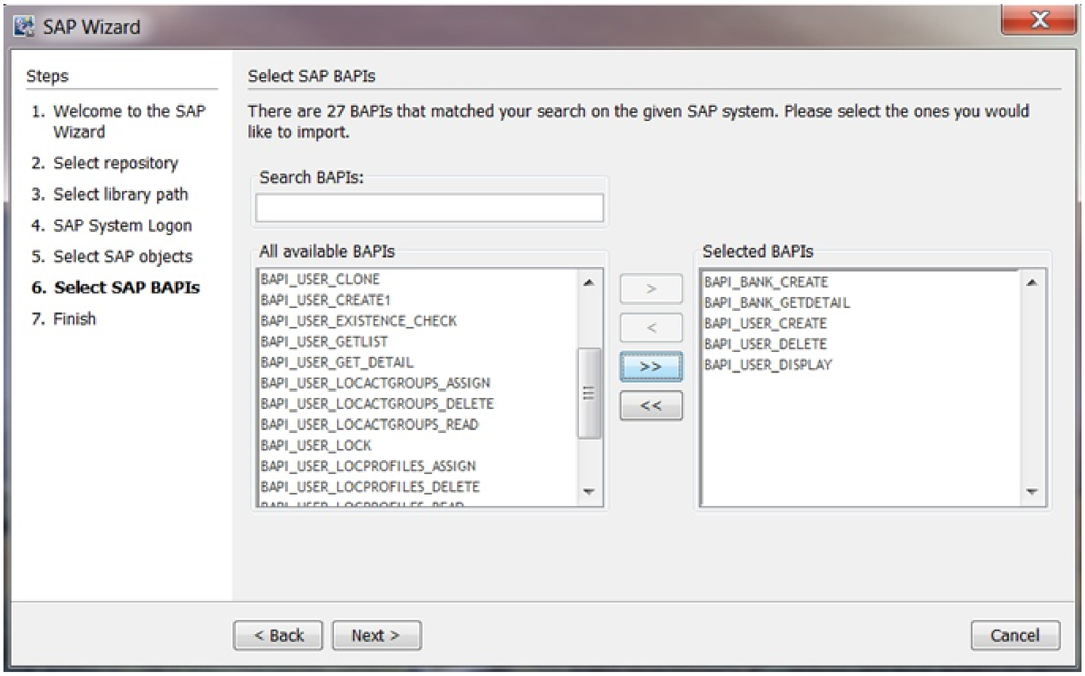
Select SAP BAPIs

There are 27 BAPIs that matched your search on the given SAP system. Please select the ones you would like to import.

All available BAPIs:

BAPI_BANK_CHANGE
BAPI_BANK_CREATE
BAPI_BANK_GETDETAIL
BAPI_BANK_GETLIST
BAPI_BANK_SAVEREPLICA
BAPI_USER_ACTGROUPS_ASSIGN
BAPI_USER_ACTGROUPS_DELETE
BAPI_USER_CHANGE
BAPI_USER_CLONE
BAPI_USER_CREATE
BAPI_USER_CREATE1

7. Select the BAPIs for which you want to generate customized flows, and then click **Next**.



SAP Wizard

Steps

1. Welcome to the SAP Wizard
2. Select repository
3. Select library path
4. SAP System Logon
5. Select SAP objects
6. **Select SAP BAPIs**
7. Finish

Select SAP BAPIs

There are 27 BAPIs that matched your search on the given SAP system. Please select the ones you would like to import.

Search BAPIs:

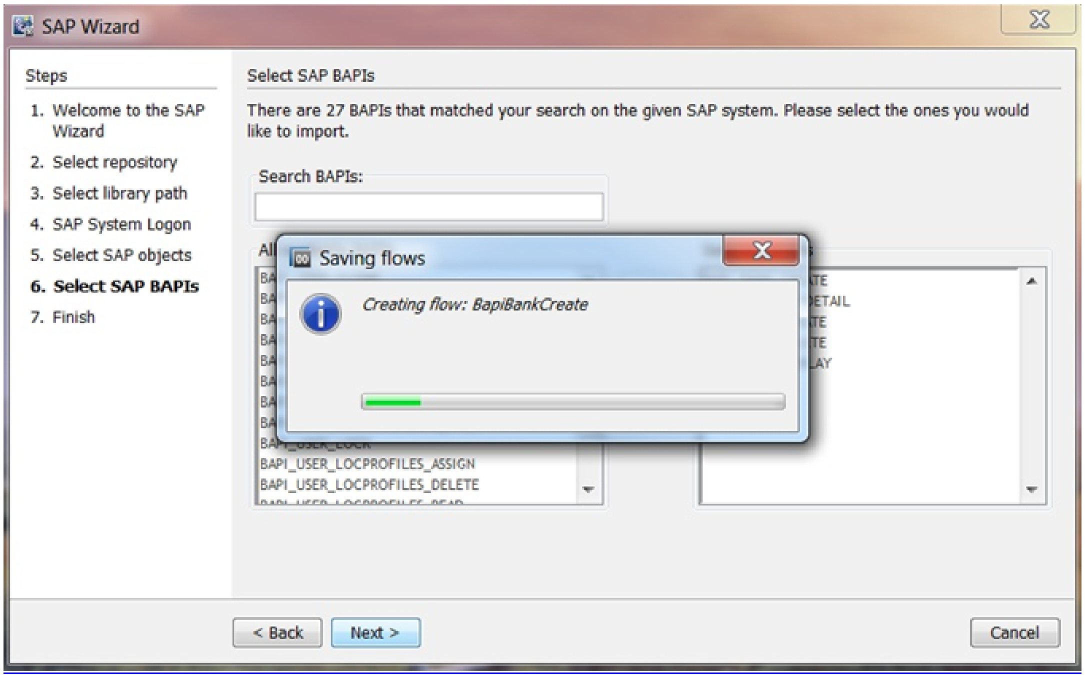
All available BAPIs

BAPI_USER_CLONE
BAPI_USER_CREATE1
BAPI_USER_EXISTENCE_CHECK
BAPI_USER_GETLIST
BAPI_USER_GET_DETAIL
BAPI_USER_LOCACTGROUPS_ASSIGN
BAPI_USER_LOCACTGROUPS_DELETE
BAPI_USER_LOCACTGROUPS_READ
BAPI_USER_LOCK
BAPI_USER_LOCPROFILES_ASSIGN
BAPI_USER_LOCPROFILES_DELETE

Selected BAPIs

BAPI_BANK_CREATE
BAPI_BANK_GETDETAIL
BAPI_USER_CREATE
BAPI_USER_DELETE
BAPI_USER_DISPLAY

Note: Generating the flows for the specified BAPIs may take a few moments, depending on the number of selected items.

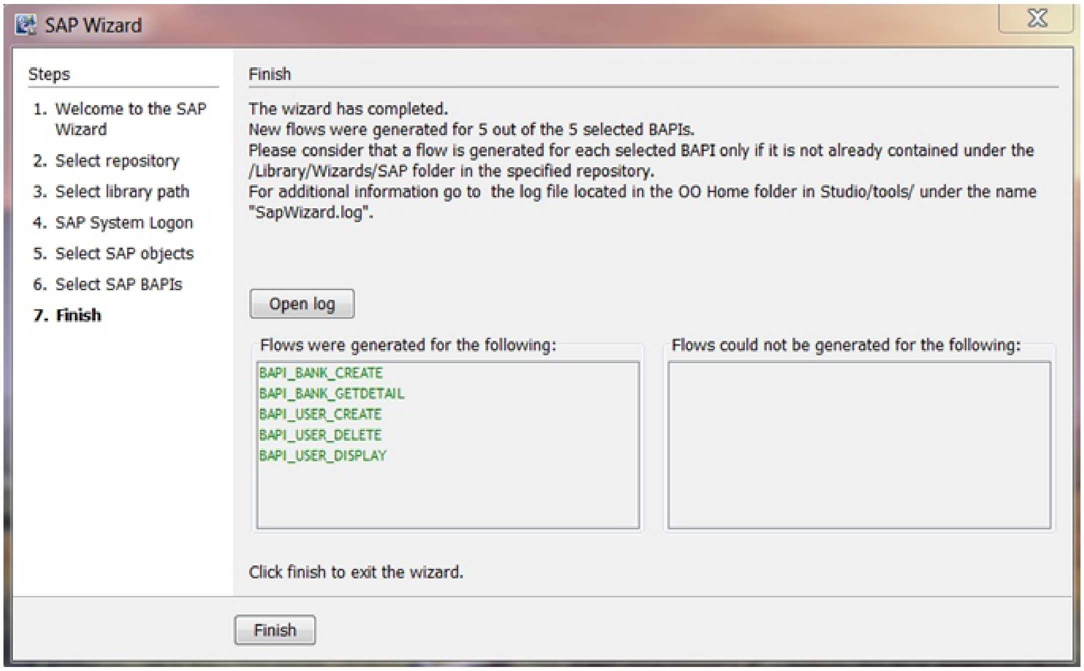


SAP Wizard

Saving flows

Creating flow: BapiBankCreate

After the flows that could be generated were created and saved in the repository, the SAP Wizard finishes. . Some of the flows may fail to be generated, due to them already existing in the repository, at the location specified in the library path selection step. For details, see "Using the SAP Wizard to Create OO Flows for Selected BAPIs" on page 33.You may choose to open the log file for additional information regarding the steps performed by the wizard.



The wizard has completed.
New flows were generated for 5 out of the 5 selected BAPIs.
Please consider that a flow is generated for each selected BAPI only if it is not already contained under the /Library/Wizards/SAP folder in the specified repository.
For additional information go to the log file located in the OO Home folder in Studio/tools/ under the name "SapWizard.log".

Flows were generated for the following:
BAPI_BANK_CREATE
BAPI_BANK_GETDETAIL
BAPI_USER_CREATE
BAPI_USER_DELETE
BAPI_USER_DISPLAY

Flows could not be generated for the following:

Click finish to exit the wizard.

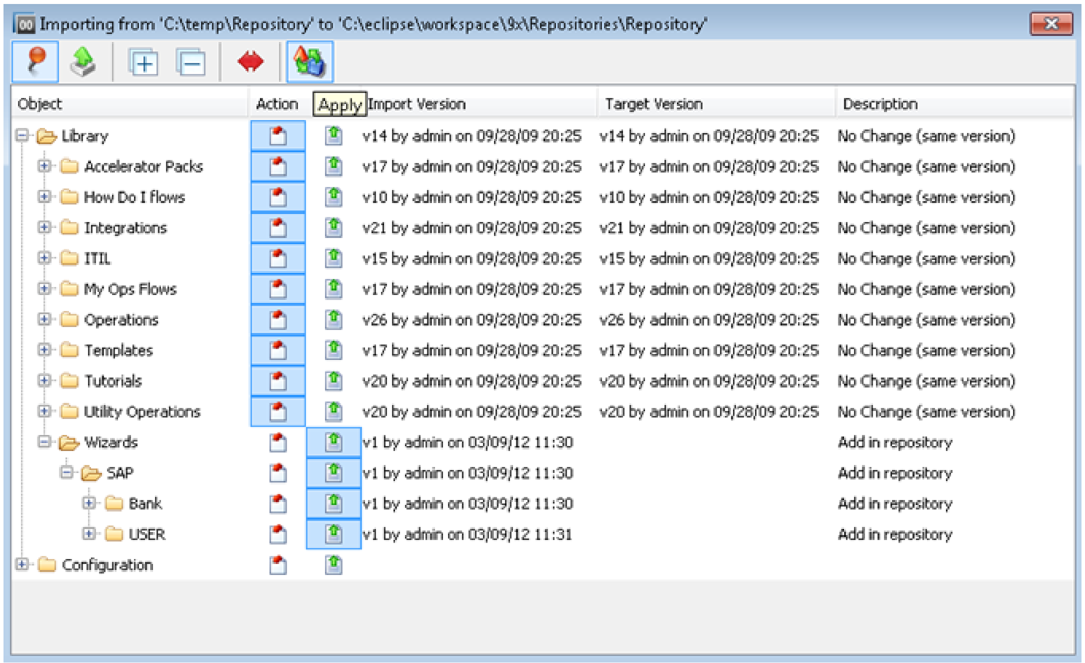
Import the Generated Flows into OO Studio

After running the SAP Wizard to create flows for specific BAPIs, import the repository into OO Studio to add the flows to the Library tree.

To import the repository into Studio:

1. In Studio, select **Import Repository** from the **Repository** menu to open the Select Repository Directory window.

   Select the repository used in the SAP Wizard (where the flows were generated), and click **Open**. The Importing from <folder> window opens.



Importing from 'C:\temp\Repository' to 'C:\eclipse\workspace\9x\Repositories\Repository'

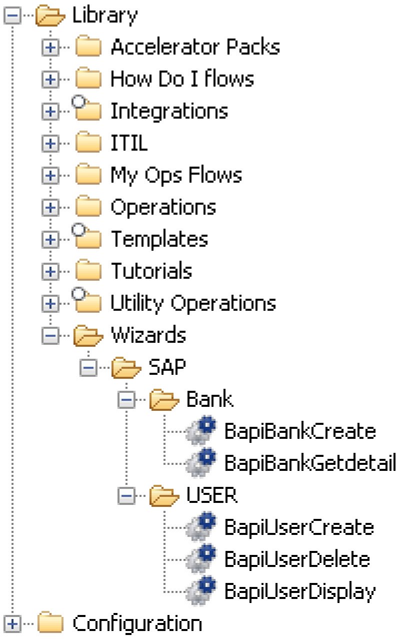
Object	Action	Apply	Import Version	Target Version	Description
Library			v14 by admin on 09/28/09 20:25	v14 by admin on 09/28/09 20:25	No Change (same version)
Accelerator Packs			v17 by admin on 09/28/09 20:25	v17 by admin on 09/28/09 20:25	No Change (same version)
How Do I flows			v10 by admin on 09/28/09 20:25	v10 by admin on 09/28/09 20:25	No Change (same version)
Integrations			v21 by admin on 09/28/09 20:25	v21 by admin on 09/28/09 20:25	No Change (same version)
ITIL			v15 by admin on 09/28/09 20:25	v15 by admin on 09/28/09 20:25	No Change (same version)
My Ops Flows			v17 by admin on 09/28/09 20:25	v17 by admin on 09/28/09 20:25	No Change (same version)
Operations			v26 by admin on 09/28/09 20:25	v26 by admin on 09/28/09 20:25	No Change (same version)
Templates			v17 by admin on 09/28/09 20:25	v17 by admin on 09/28/09 20:25	No Change (same version)
Tutorials			v20 by admin on 09/28/09 20:25	v20 by admin on 09/28/09 20:25	No Change (same version)
Utility Operations			v20 by admin on 09/28/09 20:25	v20 by admin on 09/28/09 20:25	No Change (same version)
Wizards			v1 by admin on 03/09/12 11:30		Add in repository
SAP			v1 by admin on 03/09/12 11:30		Add in repository
Bank			v1 by admin on 03/09/12 11:30		Add in repository
USER			v1 by admin on 03/09/12 11:31		Add in repository
Configuration					

2. Select **Wizards** and click **Apply** .

   If the import succeeds, the following message is displayed: **`All changes were imported successfully`**.

3. Click **OK**.

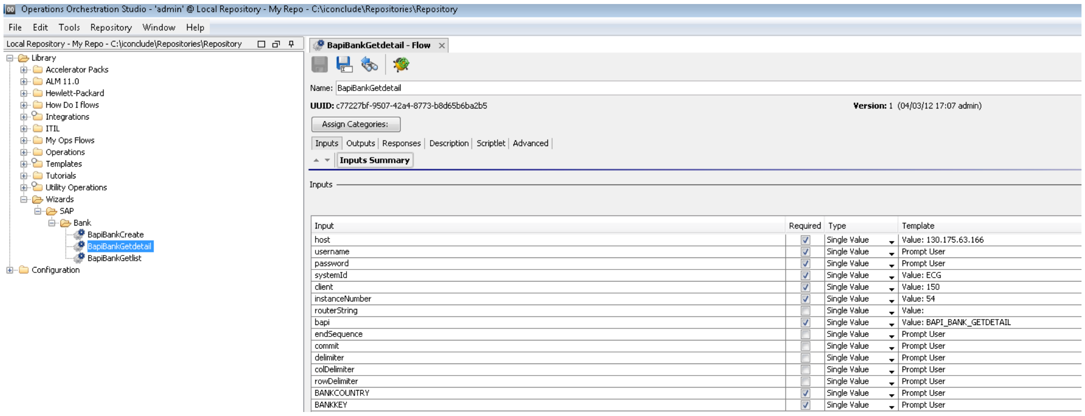
   The flows are available under the folder you selected when running the SAP Wizard. The location where the flows are stored by default is the **Library/Wizards/SAP/** folder. The generated flows are organized in folders named after the Business Objects to which they are associated.



```
Library
    Accelerator Packs
    How Do I flows
    Integrations
    ITIL
    My Ops Flows
    Operations
    Templates
    Tutorials
    Utility Operations
    Wizards
        SAP
            Bank
                BapiBankCreate
                BapiBankGetdetail
            USER
                BapiUserCreate
                BapiUserDelete
                BapiUserDisplay
Configuration
```

4. Select one of the generated flows and open the **Inputs** tab.

   All of the common inputs except for **username** and **password** are assigned from constant values according to the values of the connection parameters used in the SAP Wizard. The **bapi** input is assigned the corresponding value and all the SAP import parameters associated to the BAPI are added as inputs at flow level.



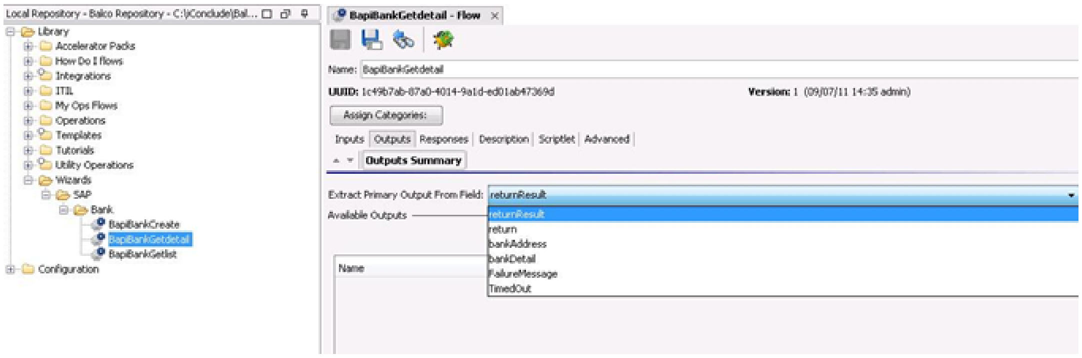
Name: BapiBankGetdetail

UUID: c77227bf-9507-42a4-8773-b8d65b6ba2b5 **Version:** 1 (04/03/12 17:07 admin)

Inputs

Input	Required	Type	Template
host	☑	Single Value	Value: 130.175.63.166
username	☑	Single Value	Prompt User
password	☑	Single Value	Prompt User
systemId	☑	Single Value	Value: ECG
client	☑	Single Value	Value: 150
instanceNumber	☑	Single Value	Value: 54
routerString	□	Single Value	Value:
bapi	☑	Single Value	Value: BAPI_BANK_GETDETAIL
endSequence	□	Single Value	Prompt User
commit	□	Single Value	Prompt User
delimiter	□	Single Value	Prompt User
colDelimiter	□	Single Value	Prompt User
rowDelimiter	□	Single Value	Prompt User
BANKCOUNTRY	☑	Single Value	Prompt User
BANKKEY	☑	Single Value	Prompt User

5. Click the **Outputs** tab to verify that the table and export parameters of the BAPI are present as results in addition to the **returnResult** result. All OO flows have the **FailureMessage** and **TimedOut** outputs. All generated flows have the **returnResult** output (which is the generic operation's result), and results corresponding to the table and export parameters of the BAPI..



Name: BapiBankGetdetail

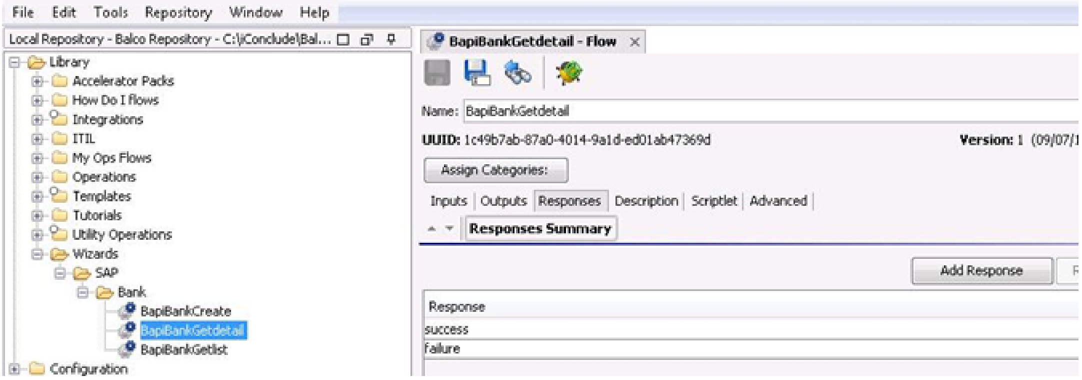
UUID: 1c49b7ab-87a0-4014-9a1d-ed01ab47369d **Version:** 1 (09/07/11 14:35 admin)

Extract Primary Output From Field: returnResult

returnResult
return
bankAddress
bankDetail
FailureMessage
TimedOut

6. Click the **Responses** tab.

   The **success** and **failure** responses of the flows are the same as in the **Invoke SAP BAPI** operation.



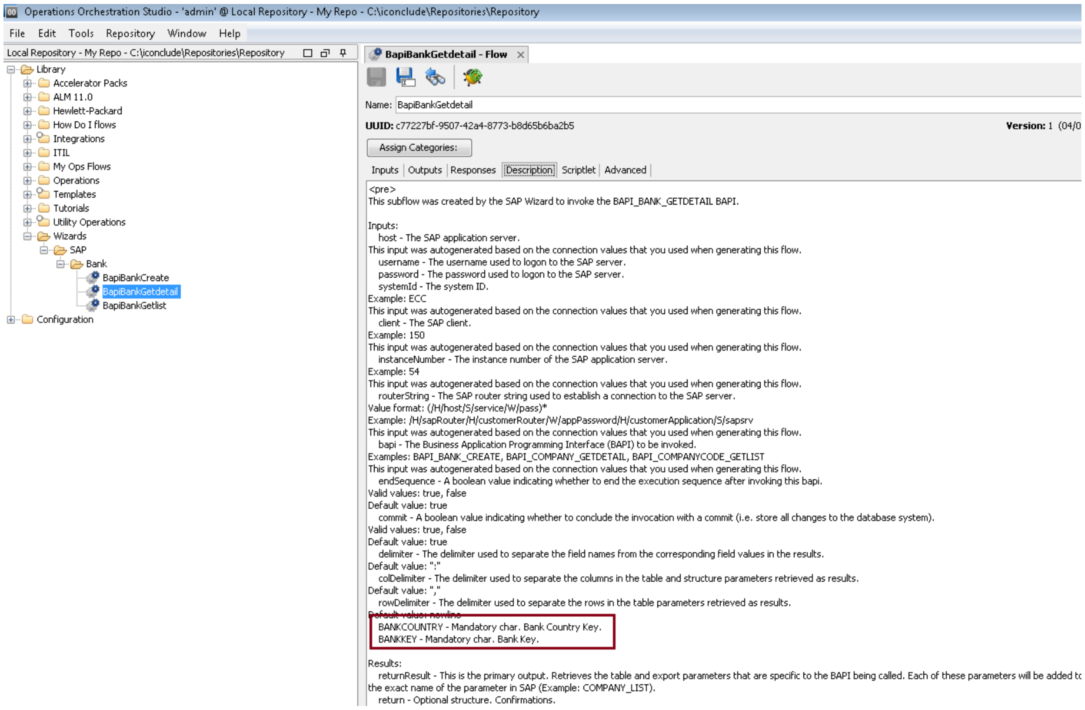
Name: BapiBankGetdetail

UUID: 1c49b7ab-87a0-4014-9a1d-ed01ab47369d **Version:** 1 (09/07/1

Response
success
failure

7. Click the **Description** tab.

   The description of the flow has been dynamically generated and includes the BAPI specific parameters in the **Inputs** and **Results** sections.



Name: BapiBankGetdetail

UUID: c77227bf-9507-42a4-8773-b8d65b6ba2b5 **Version:** 1 (04/0

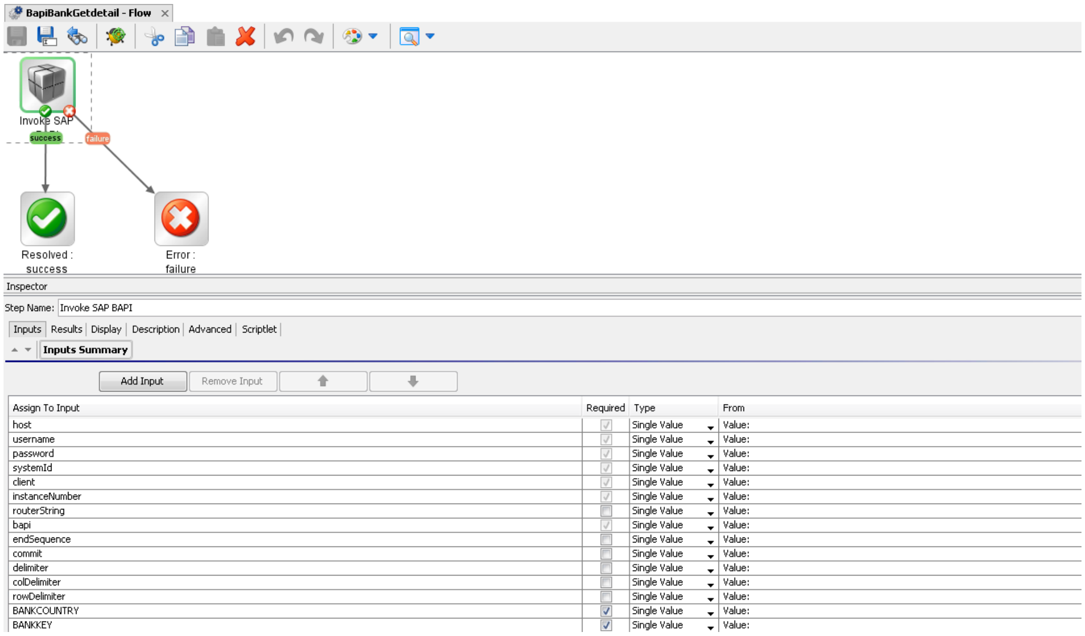
```
<pre>
This subflow was created by the SAP Wizard to invoke the BAPI_BANK_GETDETAIL BAPI.

Inputs:
   host - The SAP application server.
This input was autogenerated based on the connection values that you used when generating this flow.
   username - The username used to logon to the SAP server.
   password - The password used to logon to the SAP server.
   systemId - The system ID.
Example: ECC
This input was autogenerated based on the connection values that you used when generating this flow.
   client - The SAP client.
Example: 150
This input was autogenerated based on the connection values that you used when generating this flow.
   instanceNumber - The instance number of the SAP application server.
Example: 54
This input was autogenerated based on the connection values that you used when generating this flow.
   routerString - The SAP router string used to establish a connection to the SAP server.
Value format: (/H/host/S/service/W/pass)*
Example: /H/sapRouter/H/customerRouter/W/appPassword/H/customerApplication/S/sapsrv
This input was autogenerated based on the connection values that you used when generating this flow.
   bapi - The Business Application Programming Interface (BAPI) to be invoked.
Examples: BAPI_BANK_CREATE, BAPI_COMPANY_GETDETAIL, BAPI_COMPANYCODE_GETLIST
This input was autogenerated based on the connection values that you used when generating this flow.
   endSequence - A boolean value indicating whether to end the execution sequence after invoking this bapi.
Valid values: true, false
Default value: true
   commit - A boolean value indicating whether to conclude the invocation with a commit (i.e. store all changes to the database system).
Valid values: true, false
Default value: true
   delimiter - The delimiter used to separate the field names from the corresponding field values in the results.
Default value: ":"
   colDelimiter - The delimiter used to separate the columns in the table and structure parameters retrieved as results.
Default value: ","
   rowDelimiter - The delimiter used to separate the rows in the table parameters retrieved as results.
Default value: newline
   BANKCOUNTRY - Mandatory char. Bank Country Key.
   BANKKEY - Mandatory char. Bank Key.

Results:
   returnResult - This is the primary output. Retrieves the table and export parameters that are specific to the BAPI being called. Each of these parameters will be added to
the exact name of the parameter in SAP (Example: COMPANY_LIST).
   return - Optional structure. Confirmations.
```

8. Select the flow's step that refers to the **Invoke SAP BAPI** operation.

   The specific inputs have been added for the step as well, so their values can be passed to SAP when the BAPI is called.

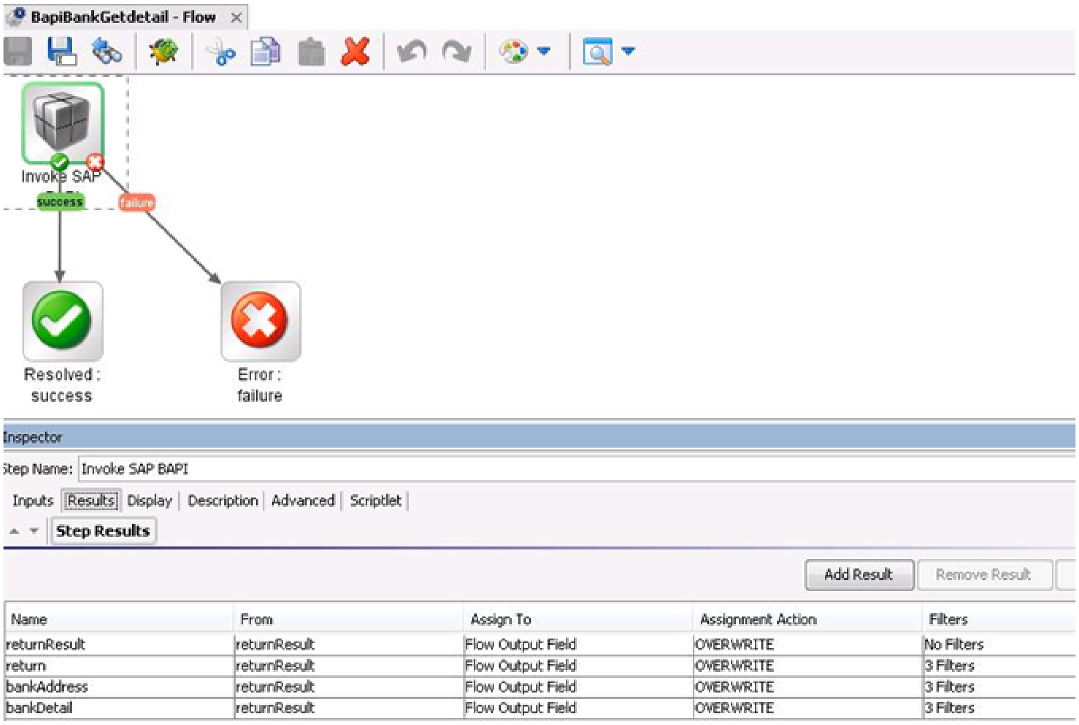


Step Name: Invoke SAP BAPI

Assign To Input	Required	Type	From
host	☑	Single Value	Value:
username	☑	Single Value	Value:
password	☑	Single Value	Value:
systemId	☑	Single Value	Value:
client	☑	Single Value	Value:
instanceNumber	☑	Single Value	Value:
routerString	□	Single Value	Value:
bapi	☑	Single Value	Value:
endSequence	□	Single Value	Value:
commit	□	Single Value	Value:
delimiter	□	Single Value	Value:
colDelimiter	□	Single Value	Value:
rowDelimiter	□	Single Value	Value:
BANKCOUNTRY	☑	Single Value	Value:
BANKKEY	☑	Single Value	Value:

9. Select the step's **Results** tab.

The BAPI-specific results have been added as flow output fields, obtained by applying filters to the raw output of the generic operation.



Step Name: Invoke SAP BAPI

Name	From	Assign To	Assignment Action	Filters
returnResult	returnResult	Flow Output Field	OVERWRITE	No Filters
return	returnResult	Flow Output Field	OVERWRITE	3 Filters
bankAddress	returnResult	Flow Output Field	OVERWRITE	3 Filters
bankDetail	returnResult	Flow Output Field	OVERWRITE	3 Filters

Chapter 6

Troubleshooting

This chapter includes:

Troubleshooting Overview

This section provides troubleshooting procedures and tools that you can use to solve problems you may encounter while using this integration. It also includes a list of the error messages you may receive while using the integration and offers descriptions and possible fixes for the errors.

General Troubleshooting Procedures and Tools

This section describes the troubleshooting procedures and tools you can use to fix problems that you may experience while using this integration.

To use the generic **Invoke SAP BAPI** operation, the sample flows and the flows created by the SAP Wizards, you must download the SAP Java Connector (JCo) from the SAP Service Marketplace at http://service.sap.com/connectors and install the connector, as stated in the generic operation's description.

> **Note:** You cannot successfully use the SAP Integration if you do not have the correct version of SAP JCo.

Error Messages

This section lists the error messages you may receive while using this integration. Each error message includes possible causes and fixes for the error.

- `java.lang.NoClassDefFoundError: com/sap/conn/jco/JCoException`

  This error occurs when you do not have SAP JCo installed, as documented in the **Invoke SAP BAPI** operation's description.

-
  ```
  java.util.concurrent.ExecutionException:
  java.lang.ExceptionInInitializerError:
   Error getting the version of the native layer: java.lang.
  ```

```
UnsatisfiedLinkError:
 no sapjco3 in java.library.path
```

This error occurs when the version of the SAP JCo you have installed is not correct. This can also happen on Windows SP2 and Windows Server 2003 R2 Enterprise SP2 when installing Java connector 3.0.4. If this is the case, install the Microsoft Visual C 2005 Service Pack 1 Redistributable Package (KB973544), that can be downloaded at http://www.microsoft.com/downloads/details.aspx?fa milyid=766a6af7-ec73-40ff-b072-9112bab119c2&displaylang=en.