Deltek
Speed. Clarity. Control.

# Costpoint Integration Console

Costpoint Integration Console

26 March 2026

While Deltek has attempted to verify that the information in this document is accurate and complete, some typographical or technical errors may exist. The recipient of this document is solely responsible for all decisions relating to or use of the information provided herein.

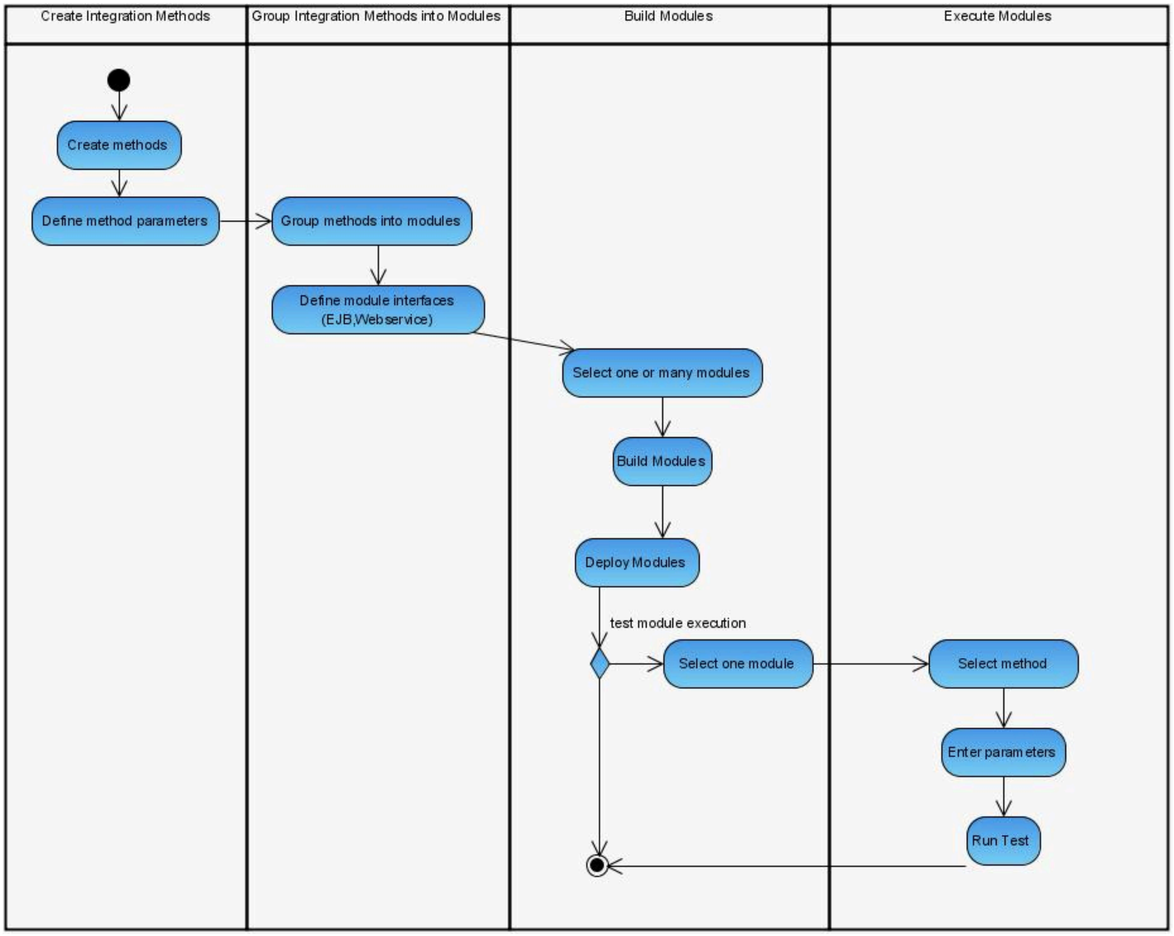
# Contents

# Costpoint Integration Console

## Costpoint Integration Console Overview

The Integration Console allows Costpoint system administrators or functional experts to expose any Costpoint processes (actions), reports, and data maintenance transactions via Web services, without writing any Java code. Currently, the console only exposes methods to be consumed by other systems. Future enhancement will include the ability to create components or facilities to consume Web service methods from other systems.

The following diagram presents a high-level overview of the process that you use to create, deploy, and test an integration method using this console.



# Procedures Overview

### Step 1: Create Method

You can create Web services for processes, reports, data export, and data import.

You can create processes and reports to be invoked synchronously or asynchronously.

Data from any maintenance/transaction applications can be exposed for export or import by selecting applications, then result sets and columns to be included.

### Step 2: Create Module

You can select methods and group them into logical modules. A module can contain a mixture of action, report, or data methods.

### Step 3: Test Module

After you create the module, it is ready to be invoked. The Integration Console provides you with the option to test a deployed module if the deployed module is on a server that is currently running. You select the modules you want to test, select the method in the module, enter the values for the parameters needed for the method, and execute the method. The method is executed, and the response is returned, exactly as if it had been executed from an outside system.

Warning: The test is executed on the Costpoint server and database. Unless you want to run this method in a "live" environment, you should run the test on either a test application server or run it against test data.

### Step 4: Export Module

Typically, you test Web services on a test server before deploying it onto production servers.

This step allows you to capture and package modules into a zip file that can be applied/deployed to another deployment.

### Step 5: Import Module

This is an optional step in which you import an integration module that has been exported from other deployments. In earlier releases of Costpoint, you had to use the Deploy Hotifix utility of the DB Wizard to import an integration module, but now you can perform this task directly from the Integration Console.

# Start the Integration Console

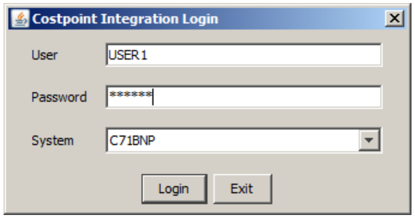
To start the Integration Console:

1. Start the Integration Console using one of the following methods:

   - Run the batch file CPWebIntegrationUtility.cmd, located with all other Costpoint command files in the \deltek\costpoint\82\bin folder.
   - From the **Start Menu** under the **Costpoint Menu** group click **Integration Console**.

2. On the Costpoint Intergration Loging dialog box, enter the following information:

   - **User**: Enter a Costpoint User ID that has rights to access the Integration Console. To allow user to access Integration Console, you need to select the **Allow Access To Integration Console Flag** option in the Maintain Users Application. Integration Console supports Database- and LDAP- (Acive Directory) based Authentication types.
   - **Password**: Enter the password associated with the User.
   - **System**: This drop-down field lists all the system names configured in Costpoint. These are the same systems that you would select if you were logging into the Costpoint application server via the Web browser.
     The default system appears in the field. Select this or another system.



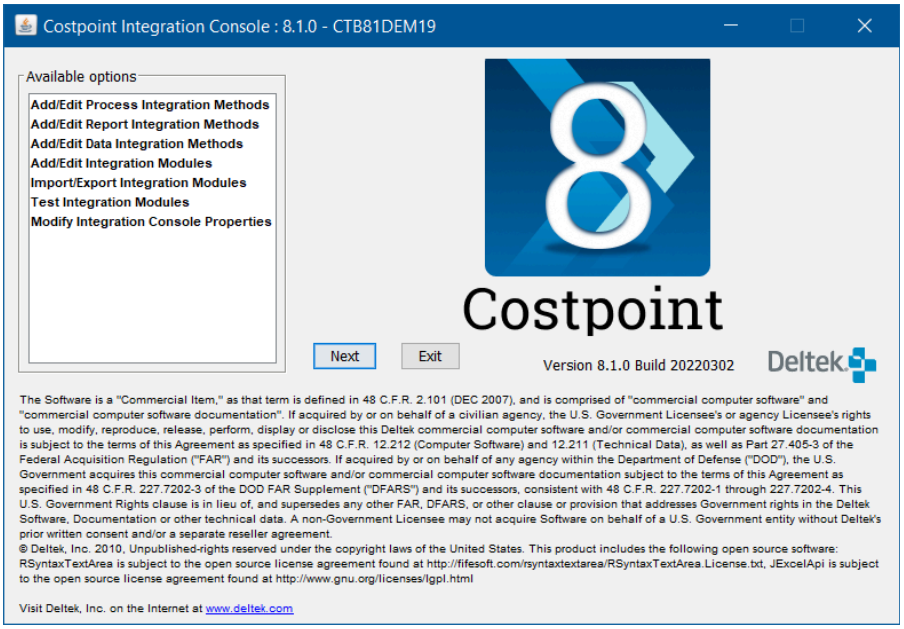
User: USER1

Password: ******

System: C71BNP

3. Click **Login** to begin your session.

   When you log in successfully, the Costpoint Integration Console screen displays. Use this screen to access all other screens.


Costpoint Integration Console : 8.1.0 - CTB81DEM19
Available options
Add/Edit Process Integration Methods
Add/Edit Report Integration Methods
Add/Edit Data Integration Methods
Add/Edit Integration Modules
Import/Export Integration Modules
Test Integration Modules
Modify Integration Console Properties
Next
Exit
8
Costpoint
Deltek

Version 8.1.0 Build 20220302

The Software is a "Commercial Item," as that term is defined in 48 C.F.R. 2.101 (DEC 2007), and is comprised of "commercial computer software" and "commercial computer software documentation". If acquired by or on behalf of a civilian agency, the U.S. Government Licensee's or agency Licensee's rights to use, modify, reproduce, release, perform, display or disclose this Deltek commercial computer software and/or commercial computer software documentation is subject to the terms of this Agreement as specified in 48 C.F.R. 12.212 (Computer Software) and 12.211 (Technical Data), as well as Part 27.405-3 of the Federal Acquisition Regulation ("FAR") and its successors. If acquired by or on behalf of any agency within the Department of Defense ("DOD"), the U.S. Government acquires this commercial computer software and/or commercial computer software documentation subject to the terms of this Agreement as specified in 48 C.F.R. 227.7202-3 of the DOD FAR Supplement ("DFARS") and its successors, consistent with 48 C.F.R. 227.7202-1 through 227.7202-4. This U.S. Government Rights clause is in lieu of, and supersedes any other FAR, DFARS, or other clause or provision that addresses Government rights in the Deltek Software, Documentation or other technical data. A non-Government Licensee may not acquire Software on behalf of a U.S. Government entity without Deltek's prior written consent and/or a separate reseller agreement.

© Deltek, Inc. 2010, Unpublished-rights reserved under the copyright laws of the United States. This product includes the following open source software: RSyntaxTextArea is subject to the open source license agreement found at http://fifesoft.com/rsyntaxtextarea/RSyntaxTextArea.License.txt, JExcelApi is subject to the open source license agreement found at http://www.gnu.org/licenses/lgpl.html

Visit Deltek, Inc. on the Internet at www.deltek.com

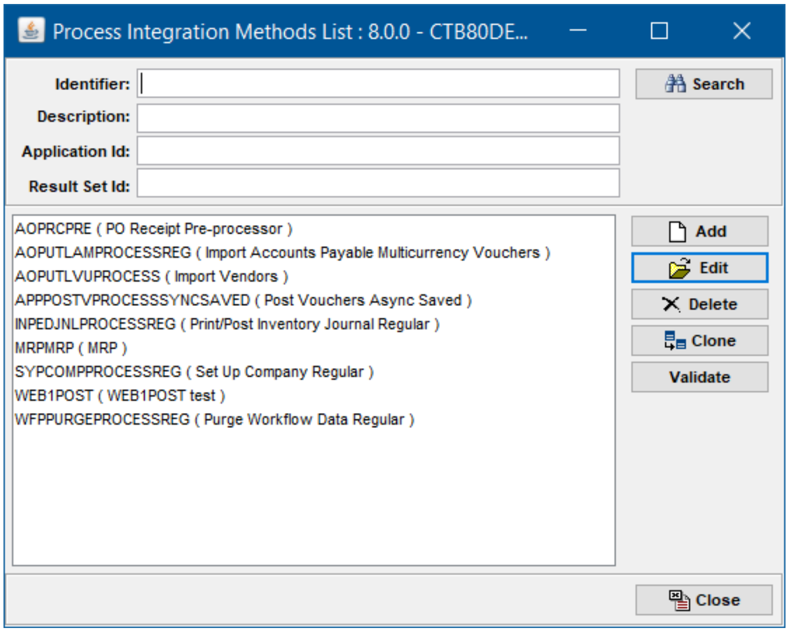
## Add/Edit Process Integration Methods

Use the **Add/Edit Process Integration Methods** option to create integration methods for Costpoint processes. Any process (action) you can run interactively from the browser can be exposed as an integration method.

### Process Integration Methods List Screen

Select **Add/Edit Process Integration Methods** to display the Process Integration Methods List screen.

On this screen, you can select existing methods to modify or create new methods.



Process Integration Methods List : 8.0.0 - CTB80DE...

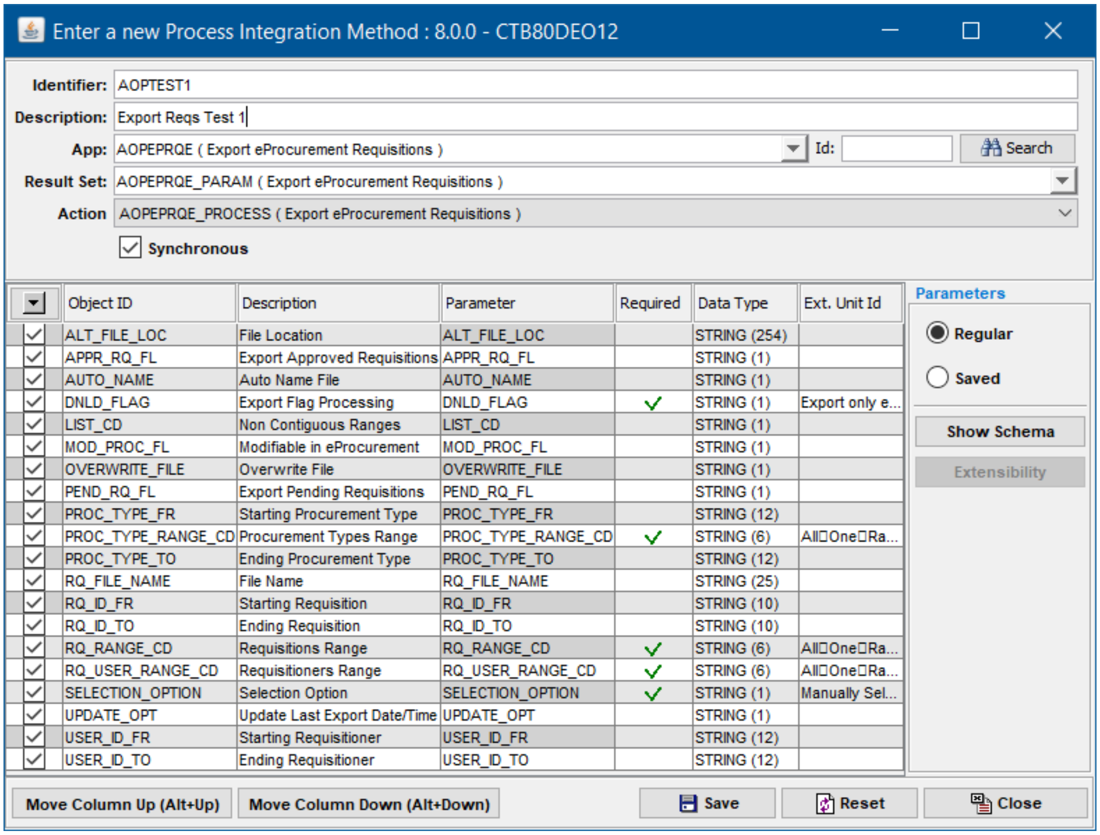
AOPRCPRE ( PO Receipt Pre-processor )
AOPUTLAMPROCESSREG ( Import Accounts Payable Multicurrency Vouchers )
AOPUTLVUPROCESS ( Import Vendors )
APPPOSTVPROCESSSYNCSAVED ( Post Vouchers Async Saved )
INPEDJNLPROCESSREG ( Print/Post Inventory Journal Regular )
MRPMRP ( MRP )
SYPCOMPPROCESSREG ( Set Up Company Regular )
WEB1POST ( WEB1POST test )
WFPPURGEPROCESSREG ( Purge Workflow Data Regular )

Use the following buttons to perform your tasks:

- Search: If the methods list is large, use the Search feature to filter the search methods based on Identifier, Description, Application Id, or Result Set Id. Enter characters in any of these fields, and click Search to display the filtered methods list. All filters use the SQL LIKE condition.
- Add: Click Add to create a new integration method. When you click this button, the Enter a New Process Integration Method screen displays. Use this screen to enter the new method.
- Edit: Click Edit to modify the selected integration method. When you click this button, the Edit Process Integration Method screen displays. Use this screen to modify the selected method.
- Delete: Select an integration method from the list, and click Delete to delete an integration method.
- Clone: Select an integration method from the list, and click Clone to create a new integration method by cloning an existing method. When you click this button, the Edit Process Integration Method screen displays. Use this screen to clone the selected method.
- Validate: Select an integration method from the list, and click Validate to validate the saved method definition against the current version of the Application metadata. It will either tell you that the validation finished successfully, or in a minority of cases when validation failed, it will suggest a course of actions to resolve the situation. You will possibly need to review and resave your method definition.
- Close: Click Close to close the Process Integration Methods List screen and return to the Costpoint Integration Console screen.

## Create a New Process Integration Method

Click Add on the Process Integration Methods List screen to create a new method.



| | |
|---|---|
| Identifier: | AOPTEST1 |
| Description: | Export Reqs Test 1 |
| App: | AOPEPRQE ( Export eProcurement Requisitions ) Id: |
| Result Set: | AOPEPRQE_PARAM ( Export eProcurement Requisitions ) |
| Action | AOPEPRQE_PROCESS ( Export eProcurement Requisitions ) |

☑ Synchronous

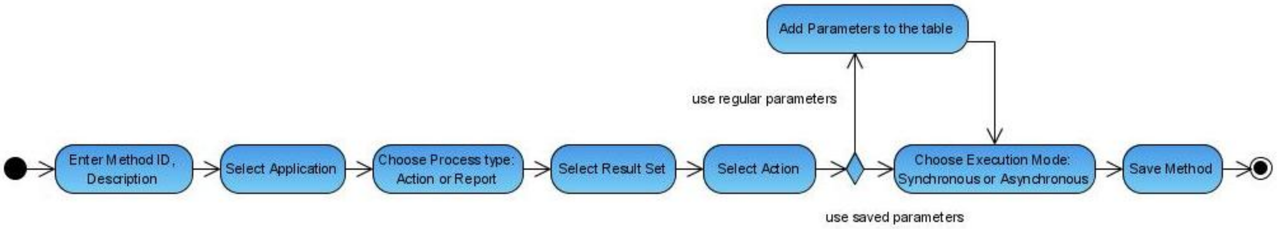
| | Object ID | Description | Parameter | Required | Data Type | Ext. Unit Id |
|---|---|---|---|---|---|---|
| ☑ | ALT_FILE_LOC | File Location | ALT_FILE_LOC | | STRING (254) | |
| ☑ | APPR_RQ_FL | Export Approved Requisitions | APPR_RQ_FL | | STRING (1) | |
| ☑ | AUTO_NAME | Auto Name File | AUTO_NAME | | STRING (1) | |
| ☑ | DNLD_FLAG | Export Flag Processing | DNLD_FLAG | ✓ | STRING (1) | Export only e... |
| ☑ | LIST_CD | Non Contiguous Ranges | LIST_CD | | STRING (1) | |
| ☑ | MOD_PROC_FL | Modifiable in eProcurement | MOD_PROC_FL | | STRING (1) | |
| ☑ | OVERWRITE_FILE | Overwrite File | OVERWRITE_FILE | | STRING (1) | |
| ☑ | PEND_RQ_FL | Export Pending Requisitions | PEND_RQ_FL | | STRING (1) | |
| ☑ | PROC_TYPE_FR | Starting Procurement Type | PROC_TYPE_FR | | STRING (12) | |
| ☑ | PROC_TYPE_RANGE_CD | Procurement Types Range | PROC_TYPE_RANGE_CD | ✓ | STRING (6) | All□One□Ra... |
| ☑ | PROC_TYPE_TO | Ending Procurement Type | PROC_TYPE_TO | | STRING (12) | |
| ☑ | RQ_FILE_NAME | File Name | RQ_FILE_NAME | | STRING (25) | |
| ☑ | RQ_ID_FR | Starting Requisition | RQ_ID_FR | | STRING (10) | |
| ☑ | RQ_ID_TO | Ending Requisition | RQ_ID_TO | | STRING (10) | |
| ☑ | RQ_RANGE_CD | Requisitions Range | RQ_RANGE_CD | ✓ | STRING (6) | All□One□Ra... |
| ☑ | RQ_USER_RANGE_CD | Requisitioners Range | RQ_USER_RANGE_CD | ✓ | STRING (6) | All□One□Ra... |
| ☑ | SELECTION_OPTION | Selection Option | SELECTION_OPTION | ✓ | STRING (1) | Manually Sel... |
| ☑ | UPDATE_OPT | Update Last Export Date/Time | UPDATE_OPT | | STRING (1) | |
| ☑ | USER_ID_FR | Starting Requisitioner | USER_ID_FR | | STRING (12) | |
| ☑ | USER_ID_TO | Ending Requisitioner | USER_ID_TO | | STRING (12) | |

Parameters

● Regular

○ Saved

The following diagram illustrates the typical steps required to create a new process integration method.



Enter Method ID, Description
Select Application
Choose Process type: Action or Report
Select Result Set
Select Action
use regular parameters
Add Parameters to the table
use saved parameters
Choose Execution Mode: Synchronous or Asynchronous
Save Method

Use these buttons at the bottom of the screen:

- **Save**: Click **Save** to save the integration method and return to the Process Integration Methods List screen.
- **Reset**: Click **Reset** to discard saved changes and load the last saved configuration.
- **Extensibility**: If your process application uses Extensibility and you want to expose the Extensibility Object IDs as part of the WS method, you need to click the **Extensibility** button and select which of Extensions assigned to this application you want to use.
- **Close**: Click **Close** to discard unsaved changes and return to the Process Integration Methods List screen.

### Enter Data for the Method

Enter data in the following fields:

- **Identifier**: Enter a unique ID for this method. Only alpha characters are allowed. The ID you specify is used as the name of the integration method. For example, if you enter **CREATEPURCHASEORDER** in the **Identifier** field, the integration method name is:
  createpurchaseorder
- **Description**: Enter a description for this method. The description is used for internal identification only and is not used elsewhere.
- **App**: Click the drop-down arrow in this field to see a list of all the Costpoint process and reporting applications to which actions are associated. Select the application that contains the end process that you want to expose for integration.
- **Id**: This field allows you to filter the list of applications based on ID. Enter the first couple of characters and click **Search**.
- **Search**: Click **Search** to filter the list of available applications.
- **Result Set**: This field automatically displays the top-level result sets that belong to the selected Costpoint application. Currently, each action application has only one result set, so this field will generally display only one result set.
- **Action**: This field contains a list of actions associated with the selected application. Most applications have only one action. The Posting application may have more than one action (Post or Print and Post). Click the drop-down arrow to select the action you want to use to create the integration method.
- **Synchronous**: Select this option to choose synchronous execution to launch a report or action. Synchronous execution requires the caller to wait until the method execution is completed before performing other Costpoint tasks. This choice is available in the interactive browser user interface as well as in this console.
  If you do not select synchronous execution, the execution will be asynchronous. Asynchronous execution returns the command to the caller immediately after receiving the instruction. The method is queued up in a Costpoint message queue to be executed there. This is the same behavior that occurs when the method is launched interactively in the browser user interface.
  A user can ask to receive an email notification when the server executes and completes the process; for this option to work, the user profile must contain an email address.
- **Parameters**: This option lets you specify the parameters for the method so that the caller of the integration method knows what to construct in the parameters when calling the method.
  There are two ways to specify how parameters can be passed:
  - **Regular** : Requires data to be specified for each individual parameter. You must construct and pass all the parameters when the method is called.
  - **Saved** : Uses existing parameters that are already saved in the Costpoint UI application.

### Use Saved Parameters

Select the **Saved** option if you want the method to use parameters that were previously saved.

In the Costpoint UI interactive mode, a user can execute an action by entering new parameters or retrieving parameters that were saved previously. Saving criteria for re-use is convenient if the action will be repeated in the future with the same parameters (for example, periodically posting vouchers).

You assign a unique parameter ID when selection parameters are saved in interactive mode.

If you use this option, the console creates a parameter in the integration method that only requires a parameter ID to be passed. At execution time, the selection parameters saved previously are automatically retrieved from the function parameter table, based on the ID.

Use this approach to expose a method that will execute on a previously saved parameter.

The following is an example of the method, using saved parameters:

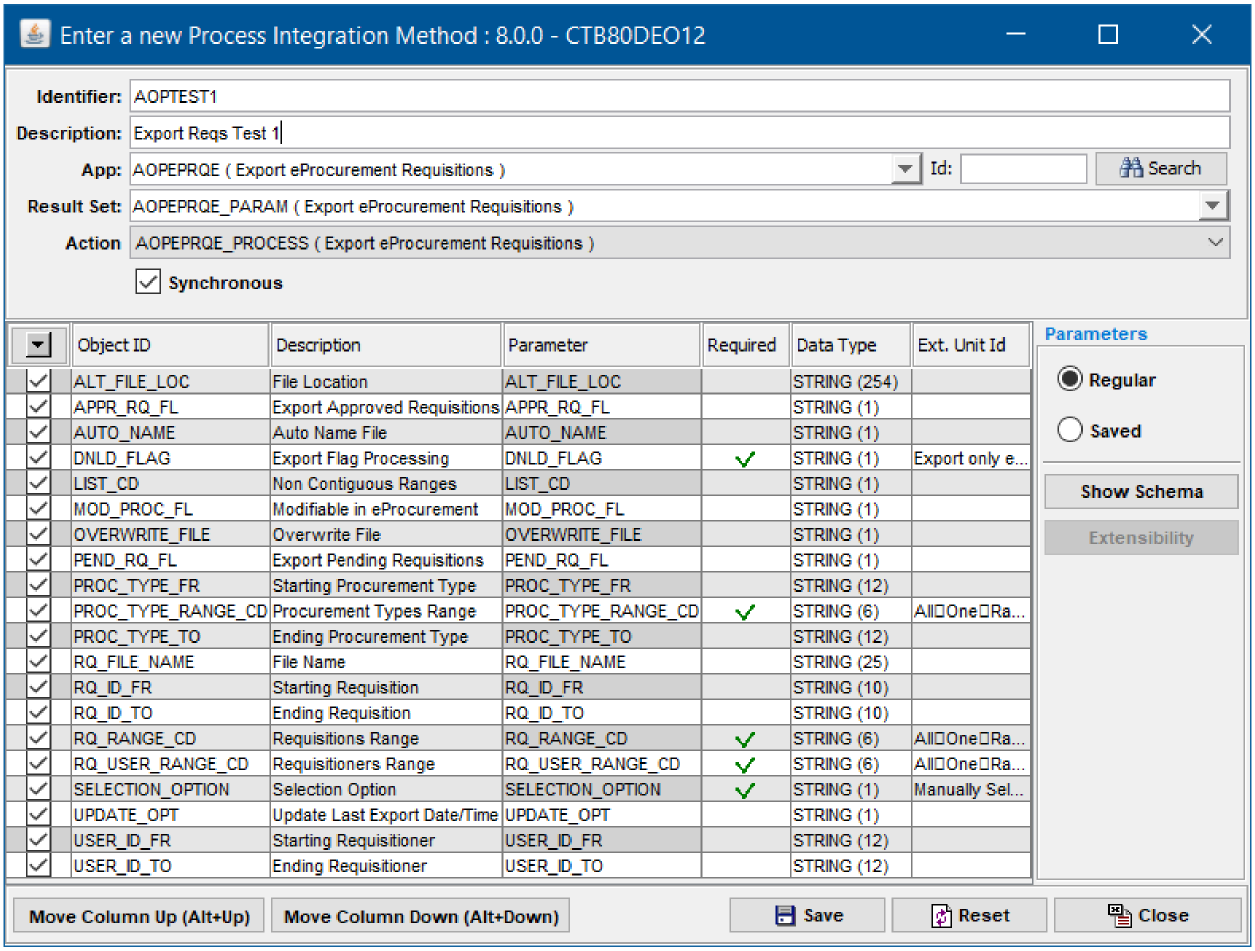
```
createpousingsavedparams ( systemName As string ,  companyId As string ,  savedParmId As string ) As MethodResponse
```

**Note:** System name and company ID are included as parameters in all methods because they are required to identify the correct system and company (similar to when you log in interactively via the browser UI).

### Use Regular Parameters

Alternatively, you can create a method that requires data for each individual parameter. You must construct and pass all the parameters when calling such a method.

If you select the **Regular** option, you will see a list of all the parameters. Select parameters using check boxes or drop-down lists.


Enter a new Process Integration Method : 8.0.0 - CTB80DEO12
Search
Parameters
Show Schema
Extensibility
Move Column Up (Alt+Up)
Move Column Down (Alt+Down)
Save
Reset
Close

Identifier: AOPTEST1

Description: Export Reqs Test 1

App: AOPEPRQE ( Export eProcurement Requisitions ) Id:

Result Set: AOPEPRQE_PARAM ( Export eProcurement Requisitions )

Action AOPEPRQE_PROCESS ( Export eProcurement Requisitions )

☑ Synchronous

● Regular

○ Saved

| | Object ID | Description | Parameter | Required | Data Type | Ext. Unit Id |
|---|---|---|---|---|---|---|
| ☑ | ALT_FILE_LOC | File Location | ALT_FILE_LOC | | STRING (254) | |
| ☑ | APPR_RQ_FL | Export Approved Requisitions | APPR_RQ_FL | | STRING (1) | |
| ☑ | AUTO_NAME | Auto Name File | AUTO_NAME | | STRING (1) | |
| ☑ | DNLD_FLAG | Export Flag Processing | DNLD_FLAG | ✓ | STRING (1) | Export only e... |
| ☑ | LIST_CD | Non Contiguous Ranges | LIST_CD | | STRING (1) | |
| ☑ | MOD_PROC_FL | Modifiable in eProcurement | MOD_PROC_FL | | STRING (1) | |
| ☑ | OVERWRITE_FILE | Overwrite File | OVERWRITE_FILE | | STRING (1) | |
| ☑ | PEND_RQ_FL | Export Pending Requisitions | PEND_RQ_FL | | STRING (1) | |
| ☑ | PROC_TYPE_FR | Starting Procurement Type | PROC_TYPE_FR | | STRING (12) | |
| ☑ | PROC_TYPE_RANGE_CD | Procurement Types Range | PROC_TYPE_RANGE_CD | ✓ | STRING (6) | All□One□Ra... |
| ☑ | PROC_TYPE_TO | Ending Procurement Type | PROC_TYPE_TO | | STRING (12) | |
| ☑ | RQ_FILE_NAME | File Name | RQ_FILE_NAME | | STRING (25) | |
| ☑ | RQ_ID_FR | Starting Requisition | RQ_ID_FR | | STRING (10) | |
| ☑ | RQ_ID_TO | Ending Requisition | RQ_ID_TO | | STRING (10) | |
| ☑ | RQ_RANGE_CD | Requisitions Range | RQ_RANGE_CD | ✓ | STRING (6) | All□One□Ra... |
| ☑ | RQ_USER_RANGE_CD | Requisitioners Range | RQ_USER_RANGE_CD | ✓ | STRING (6) | All□One□Ra... |
| ☑ | SELECTION_OPTION | Selection Option | SELECTION_OPTION | ✓ | STRING (1) | Manually Sel... |
| ☑ | UPDATE_OPT | Update Last Export Date/Time | UPDATE_OPT | | STRING (1) | |
| ☑ | USER_ID_FR | Starting Requisitioner | USER_ID_FR | | STRING (12) | |
| ☑ | USER_ID_TO | Ending Requisitioner | USER_ID_TO | | STRING (12) | |

The following is an example of the final method, using specified parameters:

```
**createpospecifyingparams** ( systemName As string , companyId As string ,  BUYE
R_RANGE_CD As string ,  NON_CONT_BUYER_FL As string ,  NON_CONT_PO_FL As string
,  NON_CONT_RQ_FL As string ,  NON_CONT_VENDOR_FL As string ,  PO_LN_SORT_CD As s
tring ,  PO_RANGE_CD As string ,  RQ_RANGE_CD As string ,  VENDOR_RANGE_CD As str
ing ) As MethodResponse
```

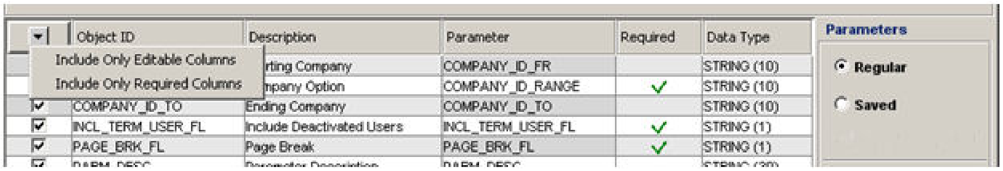
The list of parameters provides the following information:

- Object ID (Label): This column displays the internal ID and description used in the application.
- Description: This column displays the description for this parameter. The console defaults this description from the application. You can change it to a different description, as needed. (This description currently has no use, but it may be used in the future to generate documentation.)
- Parameter: This column displays the parameter name in the method. The console defaults the internal object ID into this column. You can change it to a more descriptive name using the description that is displayed in

the Description column.

- **Required**: This column indicates whether or not the parameter is required.
- **Data Type**: This column displays the parameter data type.

To make the column selection process more efficient, click the down arrow at the top left of the table.



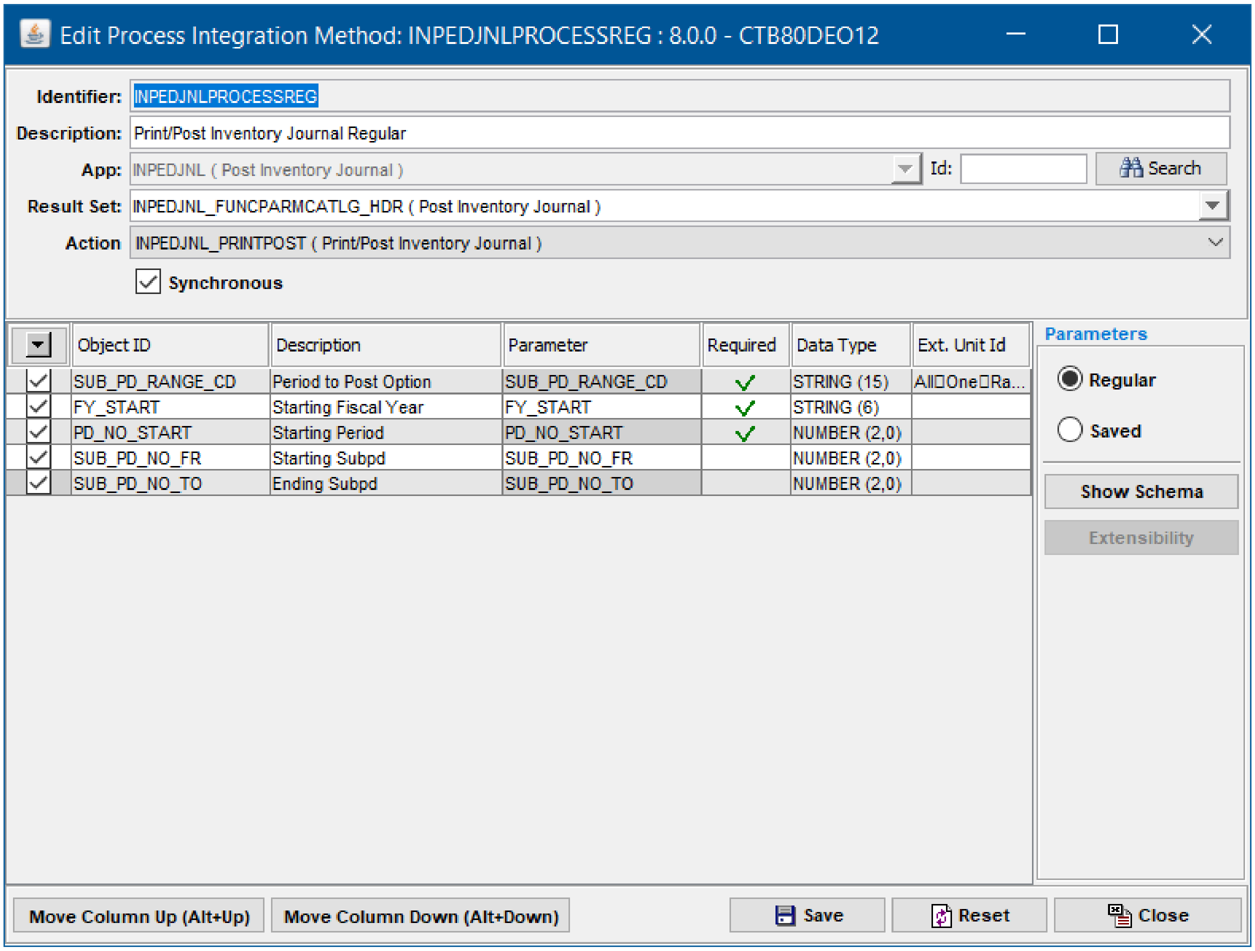
| | Object ID | Description | Parameter | Required | Data Type |
|---|---|---|---|---|---|
| | | rting Company | COMPANY_ID_FR | | STRING (10) |
| | | mpany Option | COMPANY_ID_RANGE | ✓ | STRING (10) |
| ☑ | COMPANY_ID_TO | Ending Company | COMPANY_ID_TO | | STRING (10) |
| ☑ | INCL_TERM_USER_FL | Include Deactivated Users | INCL_TERM_USER_FL | ✓ | STRING (1) |
| ☑ | PAGE_BRK_FL | Page Break | PAGE_BRK_FL | ✓ | STRING (1) |
| ☑ | PARM_DESC | Parameter Description | PARM_DESC | | STRING (30) |

Choose from these options:

- **Include Only Editable Columns**: Select this option to include only editable columns.
- **Include Only Required Columns**: Select this option to include only required columns.

### Edit a Process Integration Method

Click **Edit** on the Process Integration Methods List screen to modify an existing method. All options are the same as the options on the Enter a New Process Integration Method screen, described above. You can modify any options or parameters.



Edit Process Integration Method: INPEDJNLPROCESSREG : 8.0.0 - CTB80DEO12

Identifier: INPEDJNLPROCESSREG

Description: Print/Post Inventory Journal Regular

App: INPEDJNL ( Post Inventory Journal ) Id:

Result Set: INPEDJNL_FUNCPARMCATLG_HDR ( Post Inventory Journal )

Action INPEDJNL_PRINTPOST ( Print/Post Inventory Journal )

☑ Synchronous

| | Object ID | Description | Parameter | Required | Data Type | Ext. Unit Id |
|---|---|---|---|---|---|---|
| ☑ | SUB_PD_RANGE_CD | Period to Post Option | SUB_PD_RANGE_CD | ✓ | STRING (15) | All□One□Ra... |
| ☑ | FY_START | Starting Fiscal Year | FY_START | ✓ | STRING (6) | |
| ☑ | PD_NO_START | Starting Period | PD_NO_START | ✓ | NUMBER (2,0) | |
| ☑ | SUB_PD_NO_FR | Starting Subpd | SUB_PD_NO_FR | | NUMBER (2,0) | |
| ☑ | SUB_PD_NO_TO | Ending Subpd | SUB_PD_NO_TO | | NUMBER (2,0) | |

Parameters

● Regular

○ Saved

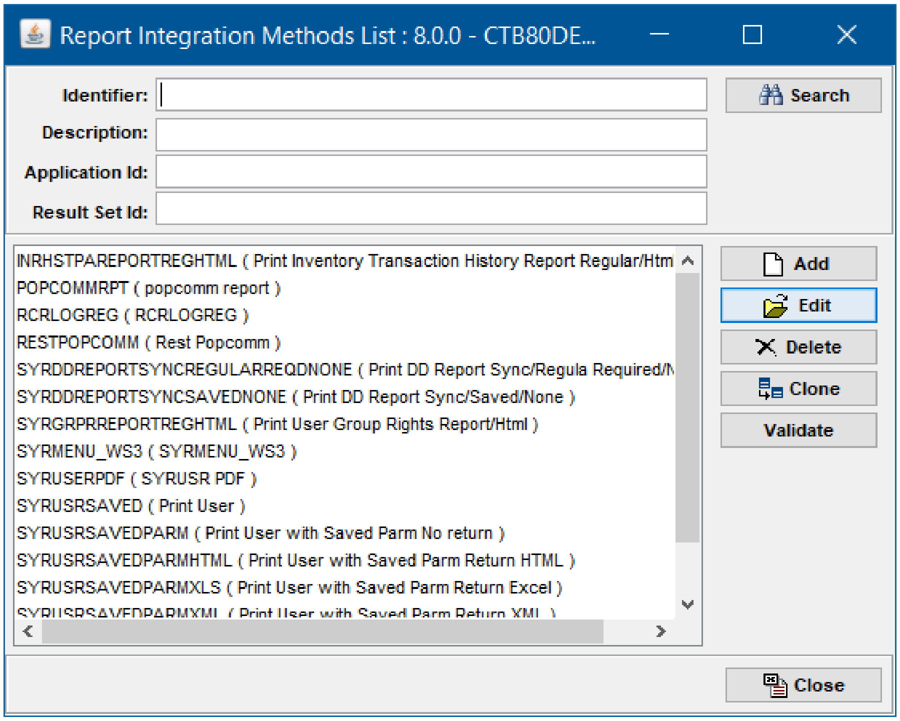
# Add/Edit Report Integration Methods

Use the **Add/Edit Report Integration Methods** option to create integration methods for Costpoint reports. Any report that you can run interactively from the browser can be created as an integration method.

When you select this option, the Report Integration Methods List screen displays.

## Report Integration Methods List

On this screen, you can select existing methods to modify, or you can create new methods.



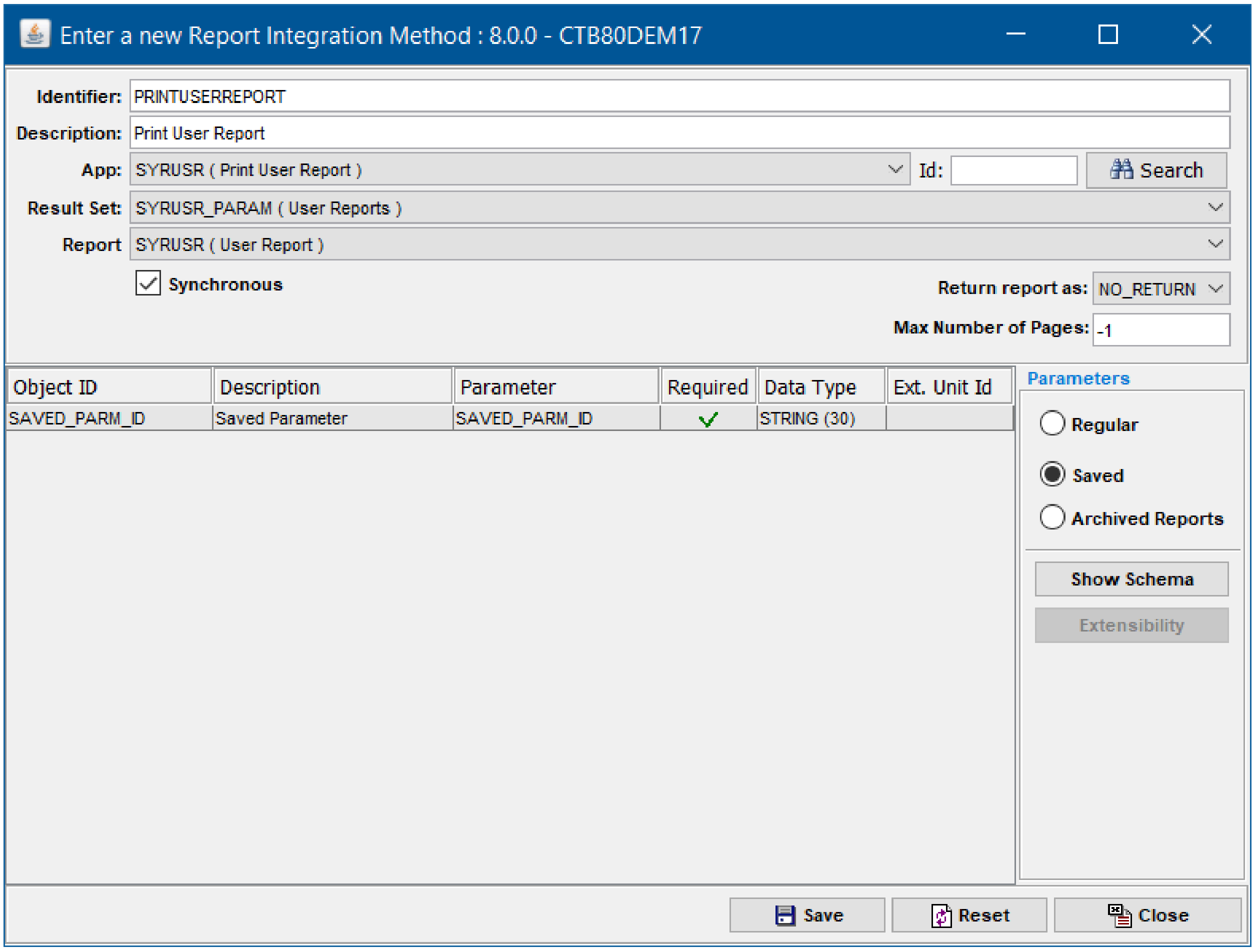
INRHSTPAREPORTREGHTML ( Print Inventory Transaction History Report Regular/Htm
POPCOMMRPT ( popcomm report )
RCRLOGREG ( RCRLOGREG )
RESTPOPCOMM ( Rest Popcomm )
SYRDDREPORTSYNCREGULARREQDNONE ( Print DD Report Sync/Regula Required/N
SYRDDREPORTSYNCSAVEDNONE ( Print DD Report Sync/Saved/None )
SYRGRPRREPORTREGHTML ( Print User Group Rights Report/Html )
SYRMENU_WS3 ( SYRMENU_WS3 )
SYRUSERPDF ( SYRUSR PDF )
SYRUSRSAVED ( Print User )
SYRUSRSAVEDPARM ( Print User with Saved Parm No return )
SYRUSRSAVEDPARMHTML ( Print User with Saved Parm Return HTML )
SYRUSRSAVEDPARMXLS ( Print User with Saved Parm Return Excel )
SYRUSRSAVEDPARMXML ( Print User with Saved Parm Return XML )

The options on this screen are the same as those on the Process Integration Methods List screen.

Attention: See the Add/Edit Process Integration Methods section for field and option descriptions.

## Enter a New Report Integration Method

Click **Add** on the Report Integration Methods List screen to create a new method.



Enter a new Report Integration Method : 8.0.0 - CTB80DEM17

|  |  |
|---|---|
| Identifier: | PRINTUSERREPORT |
| Description: | Print User Report |
| App: | SYRUSR ( Print User Report ) |
| Id: |  |
| Result Set: | SYRUSR_PARAM ( User Reports ) |
| Report | SYRUSR ( User Report ) |

☑ Synchronous

Return report as: NO_RETURN

Max Number of Pages: -1

| Object ID | Description | Parameter | Required | Data Type | Ext. Unit Id |
|---|---|---|---|---|---|
| SAVED_PARM_ID | Saved Parameter | SAVED_PARM_ID | ✓ | STRING (30) |  |

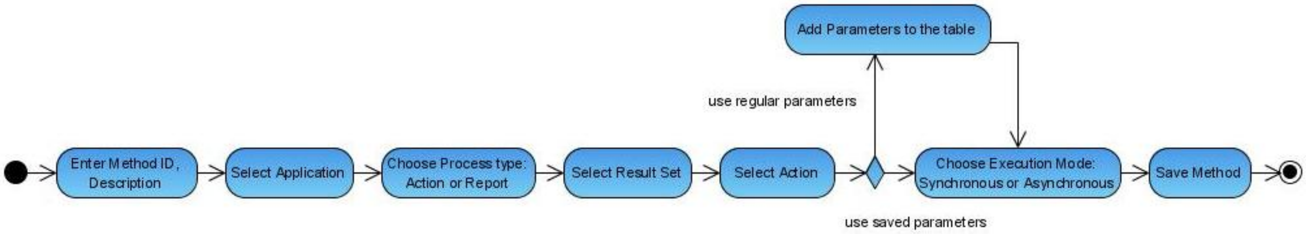
Parameters

○ Regular

● Saved

○ Archived Reports

The following diagram illustrates the typical steps required to create a new report integration method.

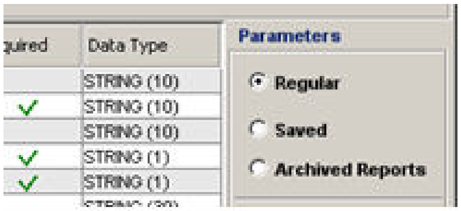


Enter Method ID, Description
Select Application
Choose Process type: Action or Report
Select Result Set
Select Action
use regular parameters
Add Parameters to the table
use saved parameters
Choose Execution Mode: Synchronous or Asynchronous
Save Method

## Enter Data for the Method

This screen has the same input fields and options as those on the Process Integration Methods List screen, plus several report-specific fields and options.

- **Archived Reports**: If you select this option, under **Parameters**, the integration method will be generated with one string parameter, Archived Report ID. At runtime, the report will not be generated, but it will be uploaded from the archive using the Archive Report ID. For this to work, the report must be generated and archived using our standard browser UI.
  **Note:** You cannot use the Archived Reports option if you have chosen Synchronous execution.



| quired | Data Type |
|---|---|
| | STRING (10) |
| ✓ | STRING (10) |
| | STRING (10) |
| ✓ | STRING (1) |
| ✓ | STRING (1) |

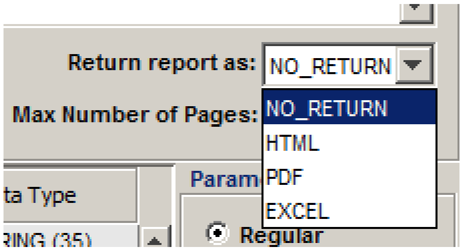
Parameters

● Regular

○ Saved

○ Archived Reports

- **Return Report As**: Use this option to specify how a report should be delivered (print to a printer, email as an attachment, download as a PDF, and so on). Different reports have different delivery options.



Return report as: NO_RETURN

Max Number of Pages:

Possible choices are:

- **NO_RETURN**: The report is not returned as part of the integration method invocation, but you can set up a saved parameter with one of many available Delivery Options in the Print Options dialog box.
  **Note:** NO_RETURN is the only option if you have chosen Synchronous execution.
- **HTML**: The report is returned as part of an HTMLReportResponse object as an HTML String along with the corresponding array of images referenced in the report.
- **PDF**: The report is returned as part of a PDFReportResponse object as a binary array that represents the report in PDF format.
- **EXCEL**: The report is returned as part of an EXCELReportResponse object as a binary array that represents the report in Excel format.
  **Note:** When a report is returned as a binary file (for example, PDF, Excel, Word, and so on), the framework uses Base64 to encode the file content. To receive the actual binary file, you will need to decode it.
- **Max Number of Pages**: This field allows you to limit the number of report pages you want to generate. Additional pages are truncated.

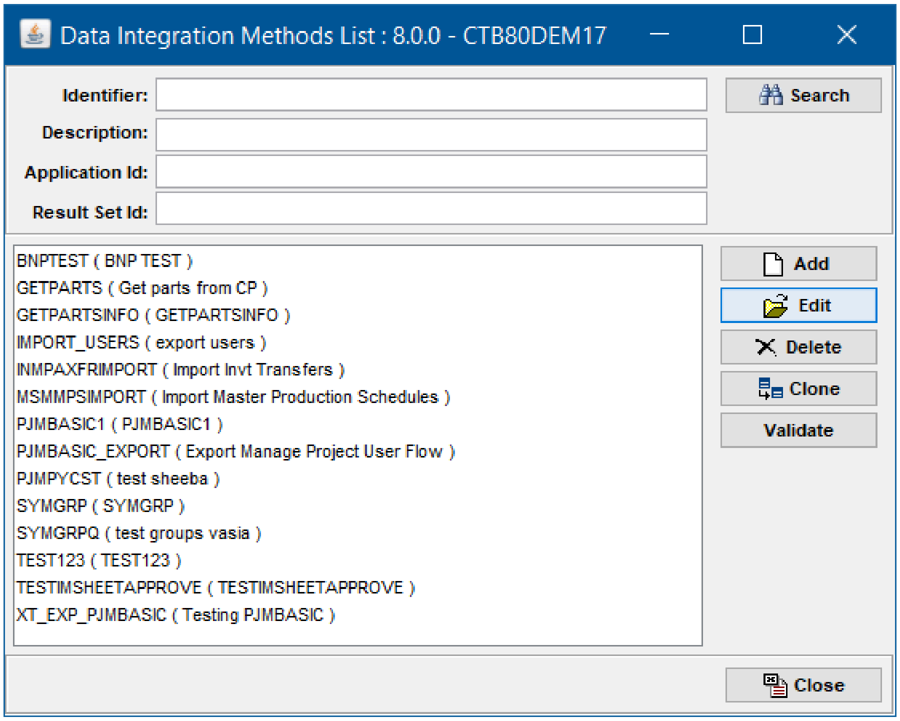
**Note:** When you run a report from the normal browser UI, the report can print only if there is sufficient information in the Print Options dialog box, which contains settings that control how the report is printed on a printer, emailed, archived, downloaded, or saved as a file.

All of these options, other than printing to a printer, require additional settings that are not currently available as parameters for an integration report method. Unless you want to print the report to the server printer, you should select the **Saved Parameters** option and specify these settings in the saved parameters.

See the Add/Edit Process Integration Methods section for information about the **Saved Parameters** option.

### Edit a Report Integration Method

Click **Edit** on the Report Integration Methods List screen to modify an existing method. All options are the same as the options on the Enter a New Report Integration Method screen. You can modify any options or parameters.



Data Integration Methods List : 8.0.0 - CTB80DEM17

Identifier:
Description:
Application Id:
Result Set Id:

BNPTEST ( BNP TEST )
GETPARTS ( Get parts from CP )
GETPARTSINFO ( GETPARTSINFO )
IMPORT_USERS ( export users )
INMPAXFRIMPORT ( Import Invt Transfers )
MSMMPSIMPORT ( Import Master Production Schedules )
PJMBASIC1 ( PJMBASIC1 )
PJMBASIC_EXPORT ( Export Manage Project User Flow )
PJMPYCST ( test sheeba )
SYMGRP ( SYMGRP )
SYMGRPQ ( test groups vasia )
TEST123 ( TEST123 )
TESTIMSHEETAPPROVE ( TESTIMSHEETAPPROVE )
XT_EXP_PJMBASIC ( Testing PJMBASIC )

## Add/Edit Data Integration (Import or Export) Methods

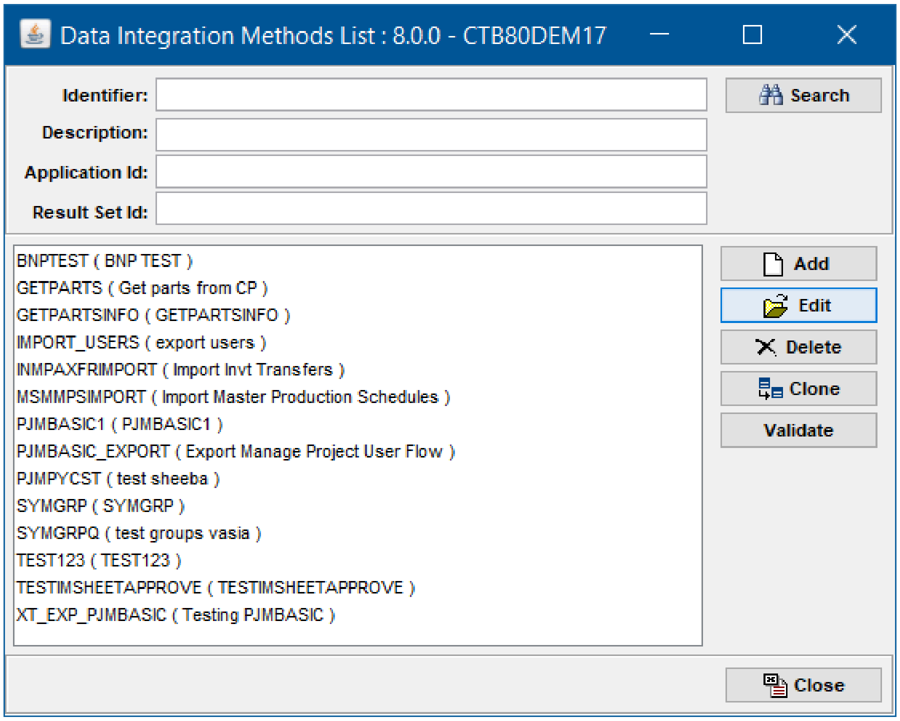
Use the **Add/Edit Data Integration Methods** option to create integration methods for Costpoint data export or import. Any application in Costpoint that displays data for viewing or accepts data for saving can be created as an integration method, including any configuration, maintenance, or transaction application.

When you select this option, the Data Integration Methods List screen displays. You can modify or delete an existing data integration method or create a new method from scratch or by cloning an existing method.

### Data Integration Methods List

Select **Add/Edit Data Integration Methods** to display the Data Integration Methods List screen.

On this screen, you can select existing methods to modify, or create new methods.



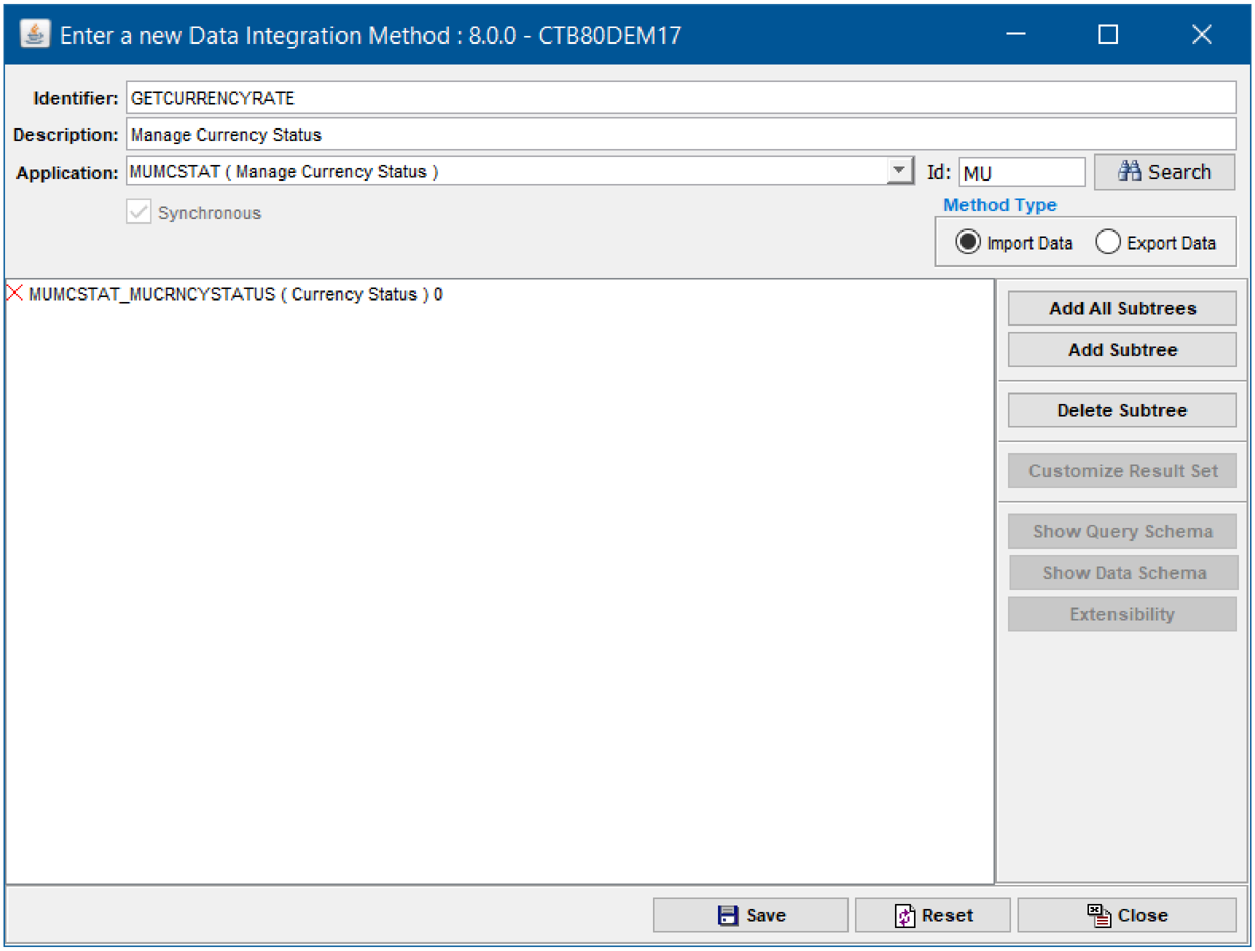
BNPTEST ( BNP TEST )
GETPARTS ( Get parts from CP )
GETPARTSINFO ( GETPARTSINFO )
IMPORT_USERS ( export users )
INMPAXFRIMPORT ( Import Invt Transfers )
MSMMPSIMPORT ( Import Master Production Schedules )
PJMBASIC1 ( PJMBASIC1 )
PJMBASIC_EXPORT ( Export Manage Project User Flow )
PJMPYCST ( test sheeba )
SYMGRP ( SYMGRP )
SYMGRPQ ( test groups vasia )
TEST123 ( TEST123 )
TESTIMSHEETAPPROVE ( TESTIMSHEETAPPROVE )
XT_EXP_PJMBASIC ( Testing PJMBASIC )

The options on this screen are the same as those on the Process Integration Methods List screen and the Report Integration Methods List screen.

Attention: See the Add/Edit Process Integration Methods section for field and option descriptions.

## Enter a New Data Integration Method

Click **Add** on the Data Integration Methods List screen to create a new method.



Enter a new Data Integration Method : 8.0.0 - CTB80DEM17

Identifier: GETCURRENCYRATE

Description: Manage Currency Status

Application: MUMCSTAT ( Manage Currency Status )

Id: MU

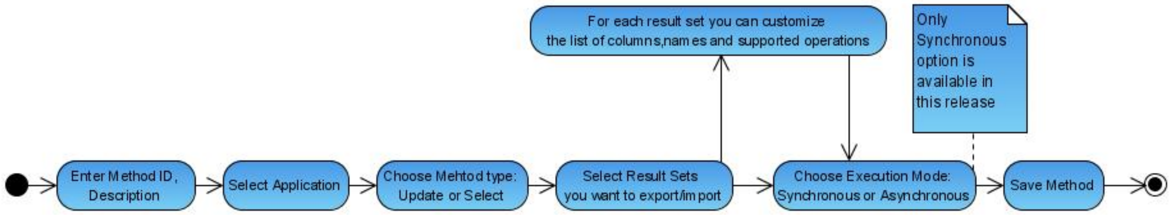
☑ Synchronous

Method Type

● Import Data ○ Export Data

MUMCSTAT_MUCRNCYSTATUS ( Currency Status ) 0

The following diagram illustrates the typical steps required to create a new data integration method.



For each result set you can customize the list of columns,nam es and supported operations
Only Synchronous option is available in this release
Enter Method ID, Description
Select Application
Choose Mehtod type: Update or Select
Select Result Sets you want to export/im port
Choose Execution Mode: Synchronous or Asynchronous
Save Method

## Enter Data for the Method

Enter data in the following fields:

- **Identifier**: Enter a unique ID for this method. Only alpha characters are allowed. The ID that you specify is used

as the name of the integration method. For example, if you enter **GETCURRENCYRATE** in the **Identifier** field, the integration method name will be:
getcurrencyrate

- **Description**: Enter a description for this method. The description is used for internal identification only and is not used elsewhere at this time.
- **Application**: Click the drop-down arrow in this field to see a list of all Costpoint applications that can export or import data. This list includes any configuration, maintenance, or transaction application.
- **Id**: This field allows you to filter the list of applications by Application ID by entering the first couple of characters and clicking Search.
- **Method type**: Select the type you want to create: **Import Data** or **Export Data**.

### Result Sets

As soon as you choose an application, the result sets that belong to the application display in the window below. These result sets are the same result sets that you see and work with when you open this application in the browser UI.

Initially, all result sets display with the **Delete** (X) icon.

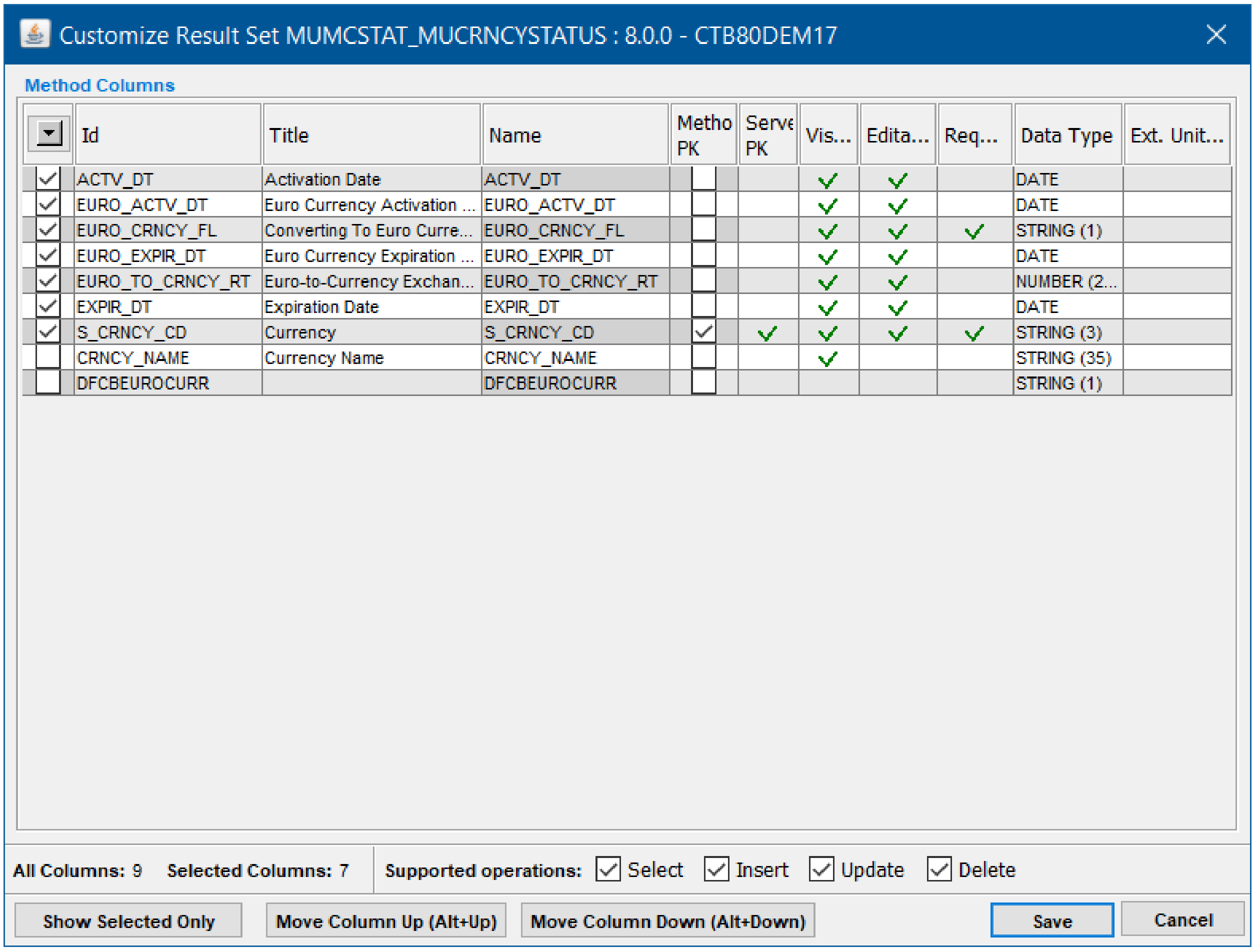
Use these buttons to control the result sets that are selected:

- **Add All Subtrees**: Click this button to add all result sets (and their children) to the tree of result sets you want to export or import. The **Delete** icon changes to a **Folder** or **Bullet** icon to signify that the result sets are now enabled.
- **Add Subtree**: Click this button to add the selected result sets (and their children) to the tree of result sets you want to export or import.
- **Delete Subtree**: Click this button to remove the selected result sets (and their children) from the tree of result sets you want to export or import.
- **Customize Result Set**: For each result set, you can select columns you want to include, customize their names, and enter extra data-specific options.

**Note:** You cannot select a subtree if its parent subtree is not selected.

### Customize the Result Set

Click the **Customize Result Set** button to display a listing of all columns in the result set. By default, all columns, both visible and hidden in the Costpoint UI, are listed here and available for selection. The column listing includes additional metadata to help you identify the columns that you want to select.



Customize Result Set MUMCSTAT_MUCRNCYSTATUS : 8.0.0 - CTB80DEM17

Method Columns

| | Id | Title | Name | Metho PK | Serve PK | Vis... | Edita... | Req... | Data Type | Ext. Unit... |
|---|---|---|---|---|---|---|---|---|---|---|
| ☑ | ACTV_DT | Activation Date | ACTV_DT | ☐ | | ✓ | ✓ | | DATE | |
| ☑ | EURO_ACTV_DT | Euro Currency Activation ... | EURO_ACTV_DT | ☐ | | ✓ | ✓ | | DATE | |
| ☑ | EURO_CRNCY_FL | Converting To Euro Curre... | EURO_CRNCY_FL | ☐ | | ✓ | ✓ | ✓ | STRING (1) | |
| ☑ | EURO_EXPIR_DT | Euro Currency Expiration ... | EURO_EXPIR_DT | ☐ | | ✓ | ✓ | | DATE | |
| ☑ | EURO_TO_CRNCY_RT | Euro-to-Currency Exchan... | EURO_TO_CRNCY_RT | ☐ | | ✓ | ✓ | | NUMBER (2... | |
| ☑ | EXPIR_DT | Expiration Date | EXPIR_DT | ☐ | | ✓ | ✓ | | DATE | |
| ☑ | S_CRNCY_CD | Currency | S_CRNCY_CD | ☑ | ✓ | ✓ | ✓ | ✓ | STRING (3) | |
| ☐ | CRNCY_NAME | Currency Name | CRNCY_NAME | ☐ | | ✓ | | | STRING (35) | |
| ☐ | DFCBEUROCURR | | DFCBEUROCURR | ☐ | | | | | STRING (1) | |

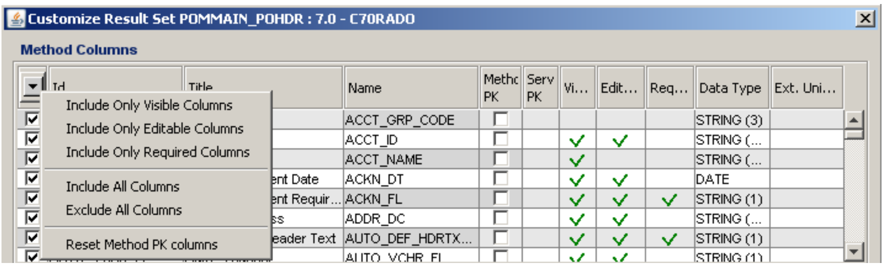
All Columns: 9 Selected Columns: 7 Supported operations: ☑ Select ☑ Insert ☑ Update ☑ Delete

The columns on this screen are:

- Id: This column displays the internal ID for the column.
- Title: This is the column title as you would see it in the Costpoint UI. If a table view title is available, it is displayed. If not, the form view title is displayed.
- Name: This column shows how the column name will be displayed in the integration method. In most cases, it will be the same as the column ID. If you have columns with the same name in several result sets, the result set name is added to the column name to keep it unique across result sets. The Integration Console lets you customize a column name, which can be convenient when a column name is too long .
- Method PK: This column is available only for data import integration methods, which require that you identify the columns that are primary keys. These are the columns that uniquely identify a row in the result set so that the integration method can properly perform the UPDATE, SELECT or DELETE operation. The exceptions to this rule are special types of result sets (Filter and Single Row ) that can only have one row. In this case you don’t have to specify method PK columns.
- Server PK: This column shows whether the column was defined as a primary key column.
- Visible: This column shows whether the column is visible in the Costpoint UI.

- **Editable**: This column shows whether the column is editable in the Costpoint UI.
- **Required**: This column shows whether the column is required in the Costpoint UI.
- **Data Type**: This column shows the column type.

### Filtering Columns for Selection


Customize Result Set POMMAIN_POHDR : 7.0 - C70RADO

Method Columns

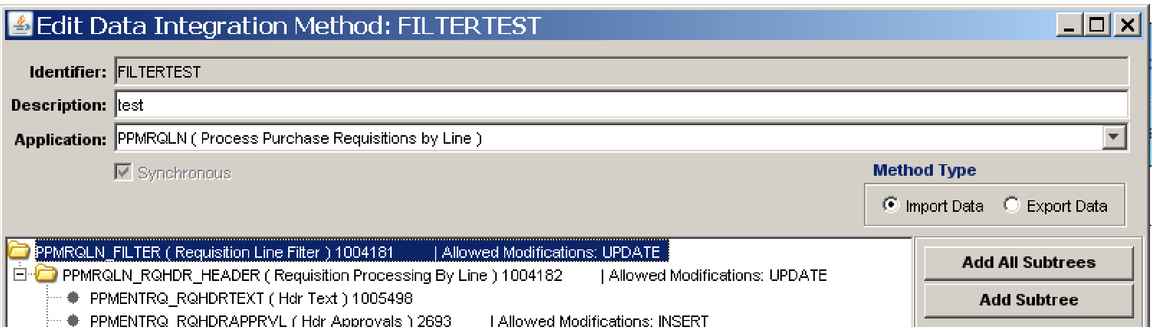
| | Id | Title | Name | Methc PK | Serv PK | Vi... | Edit... | Req... | Data Type | Ext. Uni... |
|---|---|---|---|---|---|---|---|---|---|---|
| ☑ | | | ACCT_GRP_CODE | ☐ | | | | | STRING (3) | |
| ☑ | | | ACCT_ID | ☐ | | ✓ | ✓ | | STRING (... | |
| ☑ | | | ACCT_NAME | ☐ | | ✓ | | | STRING (... | |
| ☑ | | ent Date | ACKN_DT | ☐ | | ✓ | ✓ | | DATE | |
| ☑ | | ent Requir... | ACKN_FL | ☐ | | ✓ | ✓ | ✓ | STRING (1) | |
| ☑ | | ss | ADDR_DC | ☐ | | ✓ | ✓ | | STRING (... | |
| ☑ | | eader Text | AUTO_DEF_HDRTX... | ☐ | | ✓ | ✓ | ✓ | STRING (1) | |
| ☑ | | | AUTO_VCHR_FL | ☐ | | ✓ | ✓ | | STRING (1) | |

Include Only Visible Columns
Include Only Editable Columns
Include Only Required Columns
Include All Columns
Exclude All Columns
Reset Method PK columns

To make it easier to select columns for inclusion in the method, click the down arrow to display some filtering options:

- **Include Only Visible Columns**: This option selects all columns that are visible in the Costpoint UI. This option is convenient for data export methods.
- **Include Only Editable Columns**: This option selects all columns that are editable in the Costpoint UI. This option is convenient for data import methods.
- **Include Only Required Columns**: This option selects all columns that are required in the Costpoint UI. This option is convenient for data import methods.
- **Include All Columns**: This option selects all columns.
- **Exclude All Columns**: This option deselects all columns.
- **Reset Method PK Columns**: This option puts a checkmark in the **Method PK** field for any column defined as a Server PK. This option is only available for data import methods. It usually selects the proper **Method PK** fields. However, in some rare cases, no columns exist with a Server PK label. If this occurs, select those columns that are logical keys for this result set.
  **Attention:** You may not always be able to identify logical keys. For help identifying logical key columns, see the *Deltek Costpoint Data Dictionary*.
- **Show Selected Only/Show All Columns**: This option controls the display of columns on the Customize Result Set screen. You can choose to show all columns or only those selected. This can be convenient when a result set has many columns.
- **Supported Operations**: Select the **Select, Insert, Update**, and/or **Delete** check boxes to choose which operations will be supported by the integration method.
  - For Data export methods, only **Select** is available.
  - For Data import methods, **Select**, **Insert**, **Update**, and **Delete** options are available if Costpoint allows these operations to take place for a given result at runtime.
  - Some operations have extra requirements. For example, you usually need to select Method PK columns if you choose the **Select**, **Update**, or **Delete** operations.

- If some operations are not allowed for a result set, the **Allowed Modifications** string is added to the end of the result set title. For example, in the method below, only the UPDATE operation is allowed for the top two result sets and only the INSERT operation is allowed for the "Hdr Approvals" result set.



Edit Data Integration Method: FILTERTEST

Identifier: FILTERTEST

Description: test

Application: PPMRQLN ( Process Purchase Requisitions by Line )

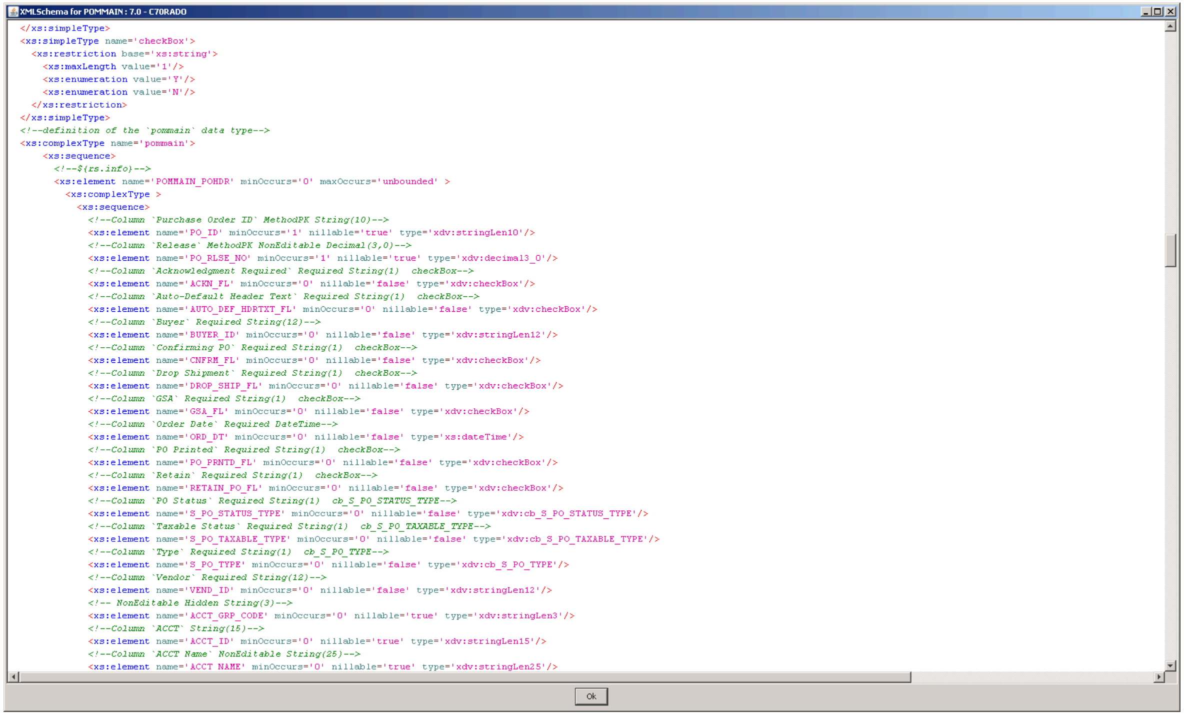
☑ Synchronous

Method Type

● Import Data ○ Export Data

```
PPMRQLN_FILTER ( Requisition Line Filter ) 1004181        | Allowed Modifications: UPDATE
  PPMRQLN_RQHDR_HEADER ( Requisition Processing By Line ) 1004182        | Allowed Modifications: UPDATE
    PPMENTRQ_RQHDRTEXT ( Hdr Text ) 1005498
    PPMENTRQ_RQHDRAPPRVL ( Hdr Approvals ) 2693        | Allowed Modifications: INSERT
```

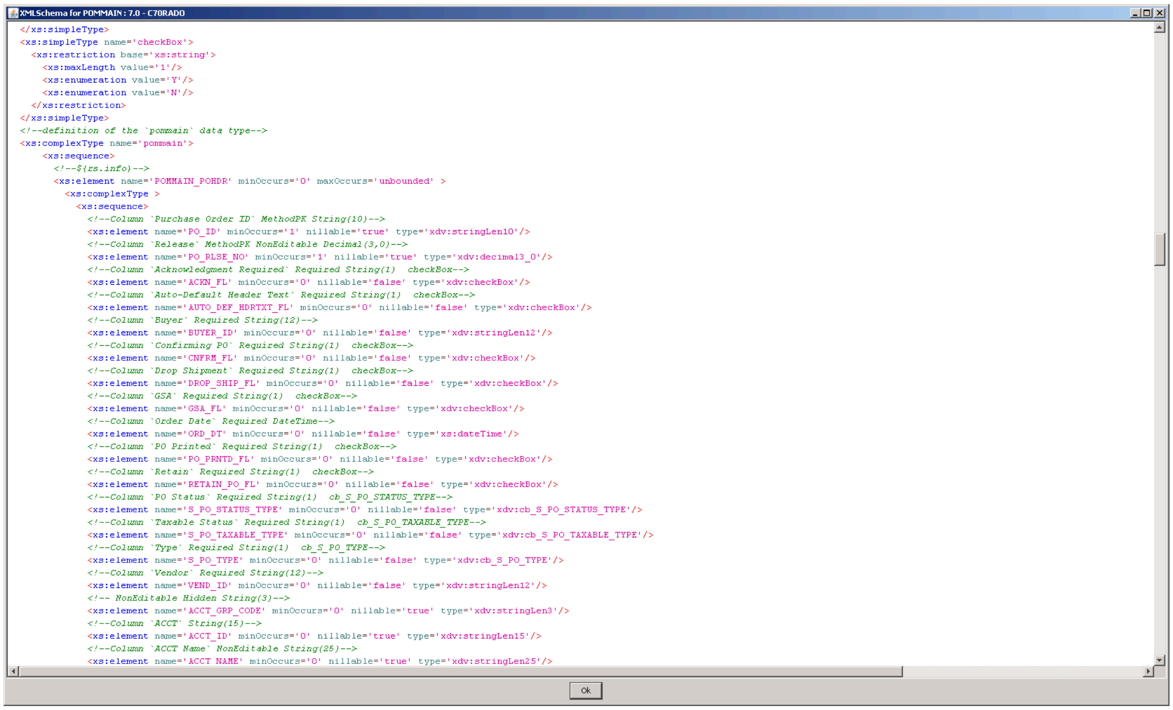
- **Show Data Schema**: Click this button to display the XML schema for the data you want to import or export.



XMLSchema for POMMAIN : 7.0 - C70RADO

```
</xs:simpleType>
<xs:simpleType name='checkBox'>
  <xs:restriction base='xs:string'>
    <xs:maxLength value='1'/>
    <xs:enumeration value='Y'/>
    <xs:enumeration value='N'/>
  </xs:restriction>
</xs:simpleType>
<!--definition of the `pommain` data type-->
<xs:complexType name='pommain'>
    <xs:sequence>
      <!--${rs.info}-->
      <xs:element name='POMMAIN_POHDR' minOccurs='0' maxOccurs='unbounded' >
        <xs:complexType >
          <xs:sequence>
            <!--Column `Purchase Order ID` MethodPK String(10)-->
            <xs:element name='PO_ID' minOccurs='1' nillable='true' type='xdv:stringLen10'/>
            <!--Column `Release` MethodPK NonEditable Decimal(3,0)-->
            <xs:element name='PO_RLSE_NO' minOccurs='1' nillable='true' type='xdv:decimal3_0'/>
            <!--Column `Acknowledgment Required` Required String(1)  checkBox-->
            <xs:element name='ACKN_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Auto-Default Header Text` Required String(1)  checkBox-->
            <xs:element name='AUTO_DEF_HDRTXT_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Buyer` Required String(12)-->
            <xs:element name='BUYER_ID' minOccurs='0' nillable='false' type='xdv:stringLen12'/>
            <!--Column `Confirming PO` Required String(1)  checkBox-->
            <xs:element name='CNFRM_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Drop Shipment` Required String(1)  checkBox-->
            <xs:element name='DROP_SHIP_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `GSA` Required String(1)  checkBox-->
            <xs:element name='GSA_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Order Date` Required DateTime-->
            <xs:element name='ORD_DT' minOccurs='0' nillable='false' type='xs:dateTime'/>
            <!--Column `PO Printed` Required String(1)  checkBox-->
            <xs:element name='PO_PRNTD_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Retain` Required String(1)  checkBox-->
            <xs:element name='RETAIN_PO_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `PO Status` Required String(1)  cb_S_PO_STATUS_TYPE-->
            <xs:element name='S_PO_STATUS_TYPE' minOccurs='0' nillable='false' type='xdv:cb_S_PO_STATUS_TYPE'/>
            <!--Column `Taxable Status` Required String(1)  cb_S_PO_TAXABLE_TYPE-->
            <xs:element name='S_PO_TAXABLE_TYPE' minOccurs='0' nillable='false' type='xdv:cb_S_PO_TAXABLE_TYPE'/>
            <!--Column `Type` Required String(1)  cb_S_PO_TYPE-->
            <xs:element name='S_PO_TYPE' minOccurs='0' nillable='false' type='xdv:cb_S_PO_TYPE'/>
            <!--Column `Vendor` Required String(12)-->
            <xs:element name='VEND_ID' minOccurs='0' nillable='false' type='xdv:stringLen12'/>
            <!-- NonEditable Hidden String(3)-->
            <xs:element name='ACCT_GRP_CODE' minOccurs='0' nillable='true' type='xdv:stringLen3'/>
            <!--Column `ACCT` String(15)-->
            <xs:element name='ACCT_ID' minOccurs='0' nillable='true' type='xdv:stringLen15'/>
            <!--Column `ACCT Name` NonEditable String(25)-->
            <xs:element name='ACCT NAME' minOccurs='0' nillable='true' type='xdv:stringLen25'/>
```

- **Show Query Schema**: This option is available only for data export methods. This screen displays the XML schema for the query conditions that you can pass to the generated data method at run-time to filter the data that you are exporting.
  For each result set, you can set one or many query conditions (which act as OR conditions in a SQL query). Each condition can contain one or many column conditions (which act as AND conditions in a SQL query) on the columns you are exporting. These columns should also be among the list of query-able columns in the original result set (those columns that you can select for query in the Costpoint UI). Each column condition is a combination of column name, relation, and value.

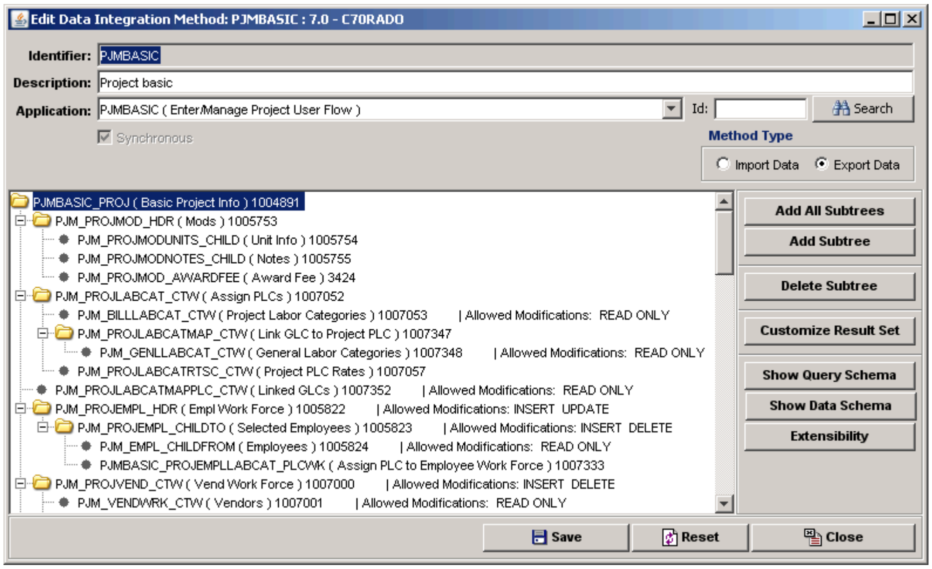


XMLSchema for POMMAIN : 7.0 - C70RADO

```
</xs:simpleType>
<xs:simpleType name='checkBox'>
  <xs:restriction base='xs:string'>
    <xs:maxLength value='1'/>
    <xs:enumeration value='Y'/>
    <xs:enumeration value='N'/>
  </xs:restriction>
</xs:simpleType>
<!--definition of the `pommain` data type-->
<xs:complexType name='pommain'>
    <xs:sequence>
      <!--${rs.info}-->
      <xs:element name='POMMAIN_POHDR' minOccurs='0' maxOccurs='unbounded' >
        <xs:complexType >
          <xs:sequence>
            <!--Column `Purchase Order ID` MethodPK String(10)-->
            <xs:element name='PO_ID' minOccurs='1' nillable='true' type='xdv:stringLen10'/>
            <!--Column `Release` MethodPK NonEditable Decimal(3,0)-->
            <xs:element name='PO_RLSE_NO' minOccurs='1' nillable='true' type='xdv:decimal3_0'/>
            <!--Column `Acknowledgment Required` Required String(1)  checkBox-->
            <xs:element name='ACKN_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Auto-Default Header Text` Required String(1)  checkBox-->
            <xs:element name='AUTO_DEF_HDRTXT_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Buyer` Required String(12)-->
            <xs:element name='BUYER_ID' minOccurs='0' nillable='false' type='xdv:stringLen12'/>
            <!--Column `Confirming PO` Required String(1)  checkBox-->
            <xs:element name='CNFRM_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Drop Shipment` Required String(1)  checkBox-->
            <xs:element name='DROP_SHIP_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `GSA` Required String(1)  checkBox-->
            <xs:element name='GSA_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Order Date` Required DateTime-->
            <xs:element name='ORD_DT' minOccurs='0' nillable='false' type='xs:dateTime'/>
            <!--Column `PO Printed` Required String(1)  checkBox-->
            <xs:element name='PO_PRNTD_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `Retain` Required String(1)  checkBox-->
            <xs:element name='RETAIN_PO_FL' minOccurs='0' nillable='false' type='xdv:checkBox'/>
            <!--Column `PO Status` Required String(1)  cb_S_PO_STATUS_TYPE-->
            <xs:element name='S_PO_STATUS_TYPE' minOccurs='0' nillable='false' type='xdv:cb_S_PO_STATUS_TYPE'/>
            <!--Column `Taxable Status` Required String(1)  cb_S_PO_TAXABLE_TYPE-->
            <xs:element name='S_PO_TAXABLE_TYPE' minOccurs='0' nillable='false' type='xdv:cb_S_PO_TAXABLE_TYPE'/>
            <!--Column `Type` Required String(1)  cb_S_PO_TYPE-->
            <xs:element name='S_PO_TYPE' minOccurs='0' nillable='false' type='xdv:cb_S_PO_TYPE'/>
            <!--Column `Vendor` Required String(12)-->
            <xs:element name='VEND_ID' minOccurs='0' nillable='false' type='xdv:stringLen12'/>
            <!-- NonEditable Hidden String(3)-->
            <xs:element name='ACCT_GRP_CODE' minOccurs='0' nillable='true' type='xdv:stringLen3'/>
            <!--Column `ACCT` String(15)-->
            <xs:element name='ACCT_ID' minOccurs='0' nillable='true' type='xdv:stringLen15'/>
            <!--Column `ACCT Name` NonEditable String(25)-->
            <xs:element name='ACCT NAME' minOccurs='0' nillable='true' type='xdv:stringLen25'/>
```

## Edit a Data Integration Method

Click **Edit** on the Data Integration Methods List screen to modify an existing method. The options on this screen are the same as the options on the Enter a New Integration Method screen. You can modify any options or parameters.



Edit Data Integration Method: PJMBASIC : 7.0 - C70RADO

**Identifier:** PJMBASIC

**Description:** Project basic

**Application:** PJMBASIC ( Enter/Manage Project User Flow )   Id:

☑ Synchronous

**Method Type**

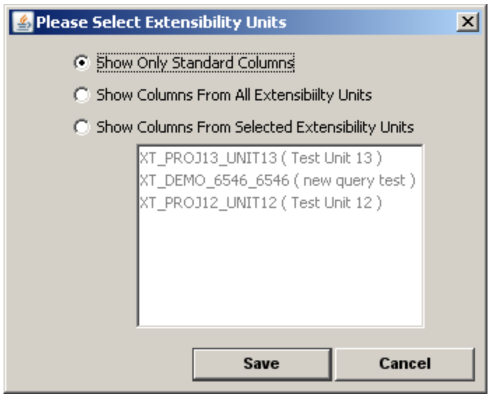
○ Import Data   ● Export Data

```
PJMBASIC_PROJ ( Basic Project Info ) 1004891
  PJM_PROJMOD_HDR ( Mods ) 1005753
      PJM_PROJMODUNITS_CHILD ( Unit Info ) 1005754
      PJM_PROJMODNOTES_CHILD ( Notes ) 1005755
      PJM_PROJMOD_AWARDFEE ( Award Fee ) 3424
  PJM_PROJLABCAT_CTW ( Assign PLCs ) 1007052
      PJM_BILLLABCAT_CTW ( Project Labor Categories ) 1007053     | Allowed Modifications:  READ ONLY
      PJM_PROJLABCATMAP_CTW ( Link GLC to Project PLC ) 1007347
          PJM_GENLLABCAT_CTW ( General Labor Categories ) 1007348     | Allowed Modifications:  READ ONLY
      PJM_PROJLABCATRTSC_CTW ( Project PLC Rates ) 1007057
  PJM_PROJLABCATMAPPLC_CTW ( Linked GLCs ) 1007352     | Allowed Modifications:  READ ONLY
  PJM_PROJEMPL_HDR ( Empl Work Force ) 1005822     | Allowed Modifications: INSERT  UPDATE
      PJM_PROJEMPL_CHILDTO ( Selected Employees ) 1005823     | Allowed Modifications: INSERT  DELETE
          PJM_EMPL_CHILDFROM ( Employees ) 1005824     | Allowed Modifications:  READ ONLY
          PJMBASIC_PROJEMPLLABCAT_PLCWK ( Assign PLC to Employee Work Force ) 1007333
  PJM_PROJVEND_CTW ( Vend Work Force ) 1007000     | Allowed Modifications: INSERT  DELETE
      PJM_VENDWRK_CTW ( Vendors ) 1007001     | Allowed Modifications:  READ ONLY
```

### Extensibility Units

If you have some extensibility units in a selected application, the **Extensibility** button becomes enabled.

When you click **Extensibility**, the Please Select Extensibility Units dialog box displays. Use this dialog box to select the Extensibility Units that you want to include in the data method.

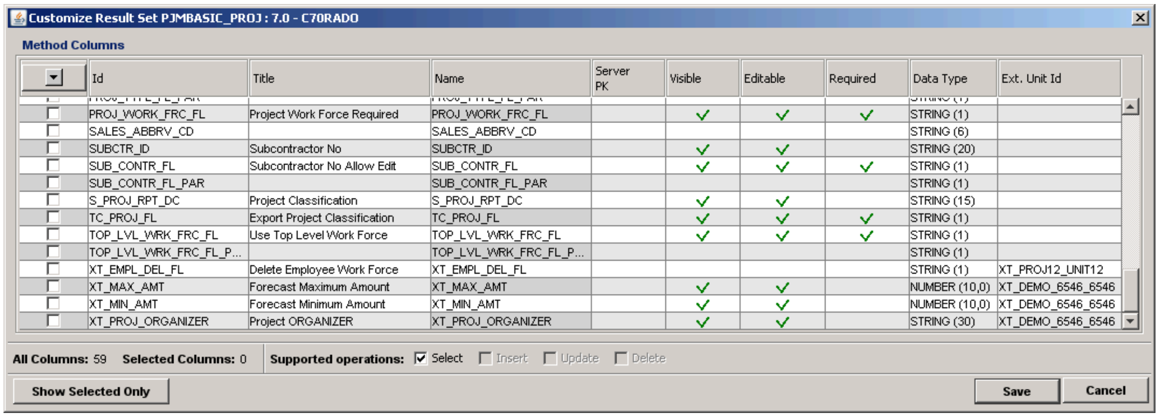


Please Select Extensibility Units

- ● Show Only Standard Columns
- ○ Show Columns From All Extensibiilty Units
- ○ Show Columns From Selected Extensibility Units

XT_PROJ13_UNIT13 ( Test Unit 13 )
XT_DEMO_6546_6546 ( new query test )
XT_PROJ12_UNIT12 ( Test Unit 12 )

When you select an Extensibility Unit, columns from this unit become available on the Customize Result Set screen. You can select the columns to be used at runtime, as you would with any regular column.



Customize Result Set PJMBASIC_PROJ : 7.0 - C70RADO

Method Columns

| | Id | Title | Name | Server PK | Visible | Editable | Required | Data Type | Ext. Unit Id |
|---|---|---|---|---|---|---|---|---|---|
| □ | PROJ_WORK_FRC_FL | Project Work Force Required | PROJ_WORK_FRC_FL | | ✓ | ✓ | ✓ | STRING (1) | |
| □ | SALES_ABBRV_CD | | SALES_ABBRV_CD | | | | | STRING (6) | |
| □ | SUBCTR_ID | Subcontractor No | SUBCTR_ID | | ✓ | ✓ | | STRING (20) | |
| □ | SUB_CONTR_FL | Subcontractor No Allow Edit | SUB_CONTR_FL | | ✓ | ✓ | ✓ | STRING (1) | |
| □ | SUB_CONTR_FL_PAR | | SUB_CONTR_FL_PAR | | | | | STRING (1) | |
| □ | S_PROJ_RPT_DC | Project Classification | S_PROJ_RPT_DC | | ✓ | ✓ | | STRING (15) | |
| □ | TC_PROJ_FL | Export Project Classification | TC_PROJ_FL | | ✓ | ✓ | ✓ | STRING (1) | |
| □ | TOP_LVL_WRK_FRC_FL | Use Top Level Work Force | TOP_LVL_WRK_FRC_FL | | ✓ | ✓ | ✓ | STRING (1) | |
| □ | TOP_LVL_WRK_FRC_FL_P... | | TOP_LVL_WRK_FRC_FL_P... | | | | | STRING (1) | |
| □ | XT_EMPL_DEL_FL | Delete Employee Work Force | XT_EMPL_DEL_FL | | | | | STRING (1) | XT_PROJ12_UNIT12 |
| □ | XT_MAX_AMT | Forecast Maximum Amount | XT_MAX_AMT | | ✓ | ✓ | | NUMBER (10,0) | XT_DEMO_6546_6546 |
| □ | XT_MIN_AMT | Forecast Minimum Amount | XT_MIN_AMT | | ✓ | ✓ | | NUMBER (10,0) | XT_DEMO_6546_6546 |
| □ | XT_PROJ_ORGANIZER | Project ORGANIZER | XT_PROJ_ORGANIZER | | ✓ | ✓ | | STRING (30) | XT_DEMO_6546_6546 |

**All Columns:** 59 **Selected Columns:** 0 **Supported operations:** ☑ Select □ Insert □ Update □ Delete

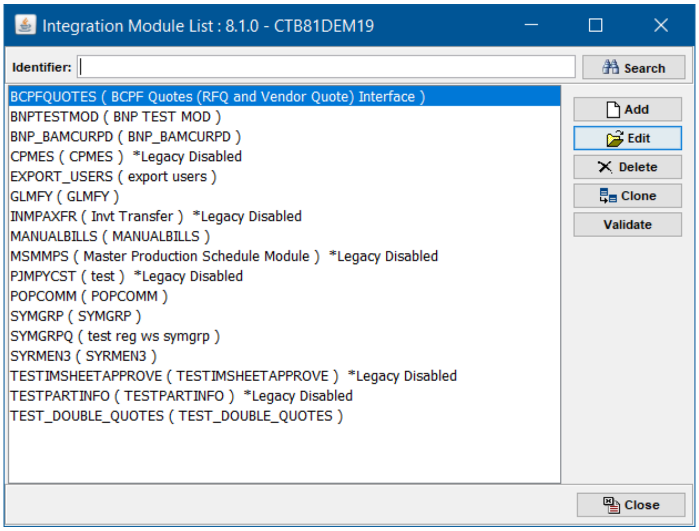
The **Ext. Unit Id** column displays the extensibility unit ID for columns that were created using the Extensibility Console. This column is empty for standard result sets.

## Add/Edit Data Integration Modules

Select **Add/Edit Data Integration Modules** to display the Integration Module List screen.

## Integration Module List Screen

On this screen, you can select existing modules to modify, or you can create new modules.

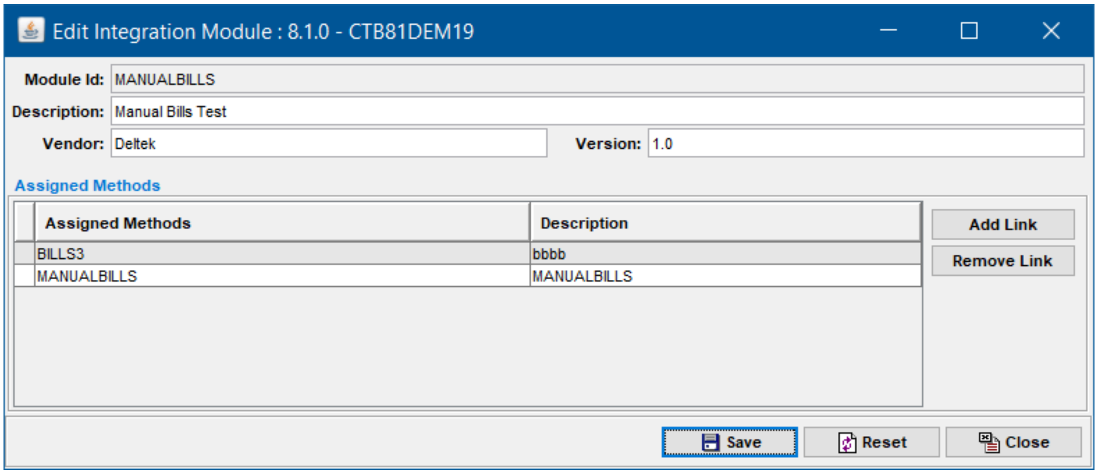


BCPFQUOTES ( BCPF Quotes (RFQ and Vendor Quote) Interface )
BNPTESTMOD ( BNP TEST MOD )
BNP_BAMCURPD ( BNP_BAMCURPD )
CPMES ( CPMES ) *Legacy Disabled
EXPORT_USERS ( export users )
GLMFY ( GLMFY )
INMPAXFR ( Invt Transfer ) *Legacy Disabled
MANUALBILLS ( MANUALBILLS )
MSMMPS ( Master Production Schedule Module ) *Legacy Disabled
PJMPYCST ( test ) *Legacy Disabled
POPCOMM ( POPCOMM )
SYMGRP ( SYMGRP )
SYMGRPQ ( test reg ws symgrp )
SYRMEN3 ( SYRMEN3 )
TESTIMSHEETAPPROVE ( TESTIMSHEETAPPROVE ) *Legacy Disabled
TESTPARTINFO ( TESTPARTINFO ) *Legacy Disabled
TEST_DOUBLE_QUOTES ( TEST_DOUBLE_QUOTES )

Use these buttons to control the modules that you select.

- Search: If the modules list is large, use the Search feature to filter the search modules based on module Identifier. Enter conditions and then click Search to display the filtered module list.
- Add: Click Add to create a new integration module. When you click this button, the Enter a New Integration Module screen displays.
- Edit: Click Edit to modify the selected integration module. When you click this button, the Edit Integration Module screen displays.
- Delete: Click Delete to delete the selected integration module.
- Clone: Select an integration module from the list and click Clone to create a new integration module by cloning an existing module. When you click this button, the Edit Integration Module screen displays. Use this screen to clone the selected module.
- Validate: Select an integration method from the list. Then click Validate to validate the saved module and linked methods definitions against the current version of the Application metadata. It will either tell you that the validation finished successfully, or in a minority of cases when validation failed, it will suggest a course of actions to resolve the situation. You will possibly need to review and resave one of the assigned method’s definitions.
- Close: Click Close to close the Integration Module List screen and return to the Costpoint Integration screen.

## Add/Edit an Integration Module



Edit Integration Module : 8.1.0 - CTB81DEM19

Module Id: MANUALBILLS

Description: Manual Bills Test

Vendor: Deltek

Version: 1.0

Assigned Methods

| Assigned Methods | Description |
|---|---|
| BILLS3 | bbbb |
| MANUALBILLS | MANUALBILLS |

### Enter/Edit Data for the Module

Enter or edit data in the following fields:

- **Module ID**: Enter a unique ID for this module. Only alpha characters are allowed.
  The ID you enter is used as part of the name of the JAR file created when the module is finally built. For example, if you enter **CURRENCY** in the **Module ID** field, the final JAR file's name will be **currencyws.jar** (the "ws" stands for Web service).
  The ID is also used as part of the WSDL file name. For example, if you enter **CURRENCY** in the **Module ID** field, the final WSDL name is **CurrencyService.WSDL**.

You can edit this field only when adding a new module.

- **Description**: Enter a description for this module. The description is used for internal identification only.
- **Vendor and Version**: These fields are used by third-party software integrators who are creating integration methods for their end users. They are used to track versions of the released module. They are used for identification only and do not affect how the integration module functions. However, Deltek highly recommend changing/incrementing the version each time the module is changed for tracking purposes.
- **Add Link**: Click this button to begin adding methods to this integration module. A new blank row displays in the **Assigned Methods** window. Click the drop-down button to select from methods (created in previous steps) to be included in the module.
- **Remove Link**: Click this button to remove an integration method from the module. (This does not delete the method itself).
- **Save Module**: Click this button to save the currently selected module.
- **Close**: Click this button to close the screen and return to the Costpoint Integration screen.

# Import/Export Integration Modules

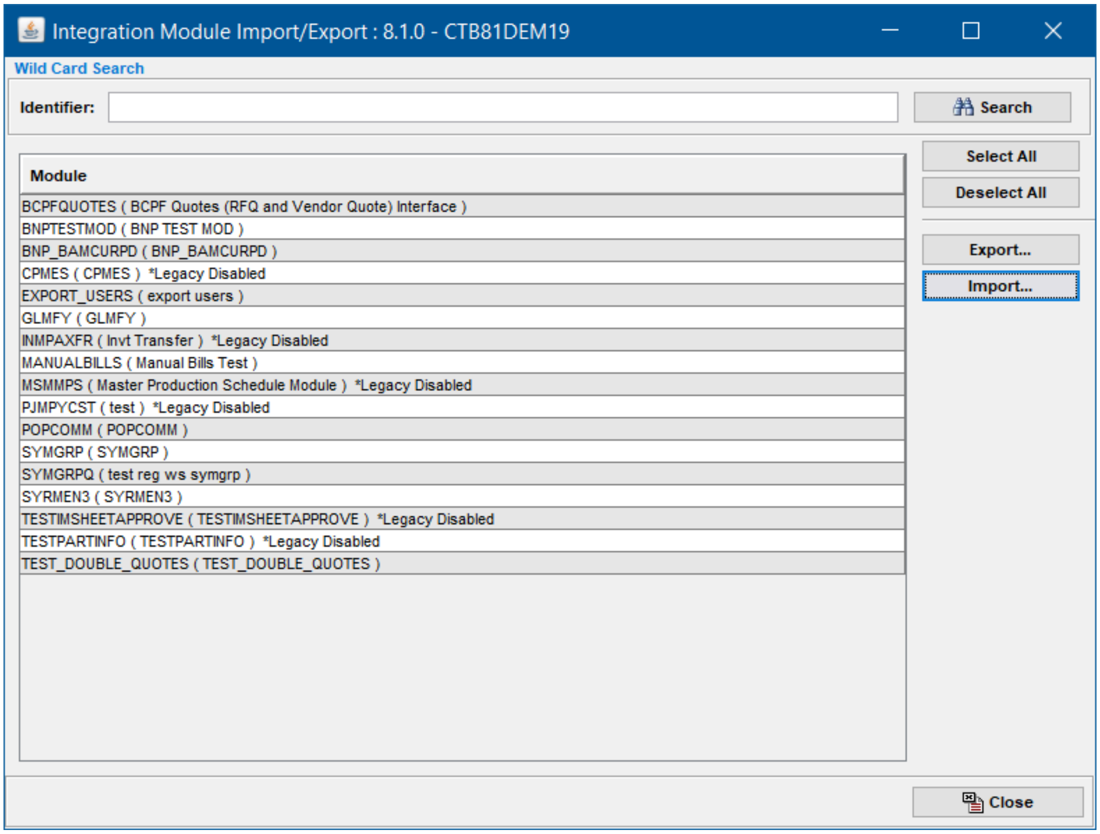
Select the **Import/Export Integration Modules** option to display the Integration Module Import/Export screen. Use this screen to select one or more modules to import or export.

Typically, you should test a Web service on a test server before deploying it onto production servers. You must import the Web service in order to execute it.

After a Web service passes initial development and testing, you can either export it from the test server and import it onto the production server or deploy it straight to the production server.

This section will help you import or export a Web service.

## Integration Module List



Module

BCPFQUOTES ( BCPF Quotes (RFQ and Vendor Quote) Interface )
BNPTESTMOD ( BNP TEST MOD )
BNP_BAMCURPD ( BNP_BAMCURPD )
CPMES ( CPMES ) *Legacy Disabled
EXPORT_USERS ( export users )
GLMFY ( GLMFY )
INMPAXFR ( Invt Transfer ) *Legacy Disabled
MANUALBILLS ( Manual Bills Test )
MSMMPS ( Master Production Schedule Module ) *Legacy Disabled
PJMPYCST ( test ) *Legacy Disabled
POPCOMM ( POPCOMM )
SYMGRP ( SYMGRP )
SYMGRPQ ( test reg ws symgrp )
SYRMEN3 ( SYRMEN3 )
TESTIMSHEETAPPROVE ( TESTIMSHEETAPPROVE ) *Legacy Disabled
TESTPARTINFO ( TESTPARTINFO ) *Legacy Disabled
TEST_DOUBLE_QUOTES ( TEST_DOUBLE_QUOTES )

This screen displays all existing modules, with their deployment status

Select one or more module from the list, or perform a search to find the modules you want to export.

Click **Select All** to select all modules or **Deselect All** to deselect all modules

To search, enter a filter, based on the module ID, in the **Identifier** field, and click **Search**. The filter is applied using the LIKE condition.

### Data about Modules

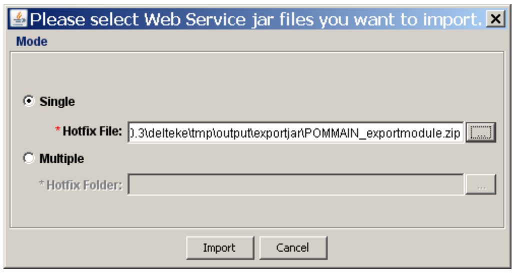
For each module on the screen, you can review the following information:

- **Module**: This is the module ID and description. If you see the words ***Legacy Disabled** next to the Module Id and Name, it means that this module was created in a prior (to Costpoint 8.1) version and is no longer supported by the product and needs to be converted into a supported format. In the majority of cases, to convert it to a supported format, opening the Module and re-saving it will be sufficient.

### Options

Use these buttons:

- **Export**: Package the module and the metadata used to create this module into an installation zip file. This file can then be installed and deployed to another system.
- **Import**: Click this button to deploy a previously exported integration module. When you click this button, a simple dialog box displays where you have to either choose the individual module that you want to import or a folder that contains several modules. Click **Import** to import them.



Please select Web Service jar files you want to import.

Mode

● Single

* Hotfix File: ).3\delteke\tmp\output\exportjar\POMMAIN_exportmodule.zip

○ Multiple

* Hotfix Folder:

- **Close**: Click this button to close the screen and return to the Costpoint Integration screen.

## Export a Module as a Hotfix

The export process captures the metadata used to create the module and its integration methods, and creates a hotfix file (in the form of a zip file) that is ready to be applied to another system. This process is designed to allow users to create and test modules in a development/testing environment before applying them in a production environment. It is also designed to allow a third-party system integrator or Deltek consulting team to develop integration modules off-site and then redeploy the modules on site.

Any module can be selected for export as long as the metadata is not missing.

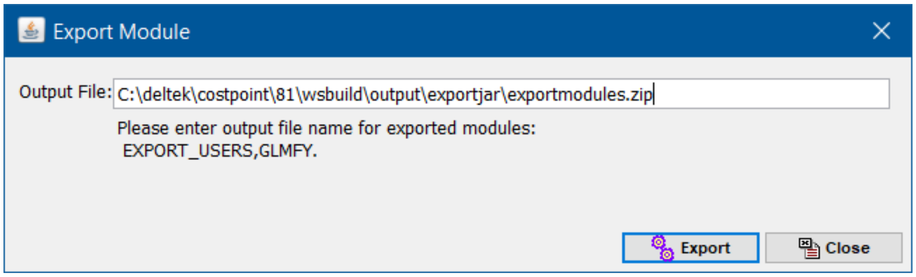
The final hotfix file (in the form of a zip file) contains a manifest file and a JAR file. The manifest file contains basic information about the JAR file. The JAR file contains the SQL scripts for the metadata and the module JAR file.

To export one or more modules:

1. Select the modules to be exported, and click **Export**.

2. On the Export dialog box, accept or override the name and location of the hotfix file.

3. Click **Export** to create the hotfix file. One hotfix file will be produced for all modules selected.



Output File: C:\deltek\costpoint\81\wsbuild\output\exportjar\exportmodules.zip

Please enter output file name for exported modules:
EXPORT_USERS,GLMFY.

## Test an Integration Module

After you define the module, it is immediately available to test. Typically, you deploy the module in a test environment, perform tests, and then deploy the module on a production server.

The main testing steps are:

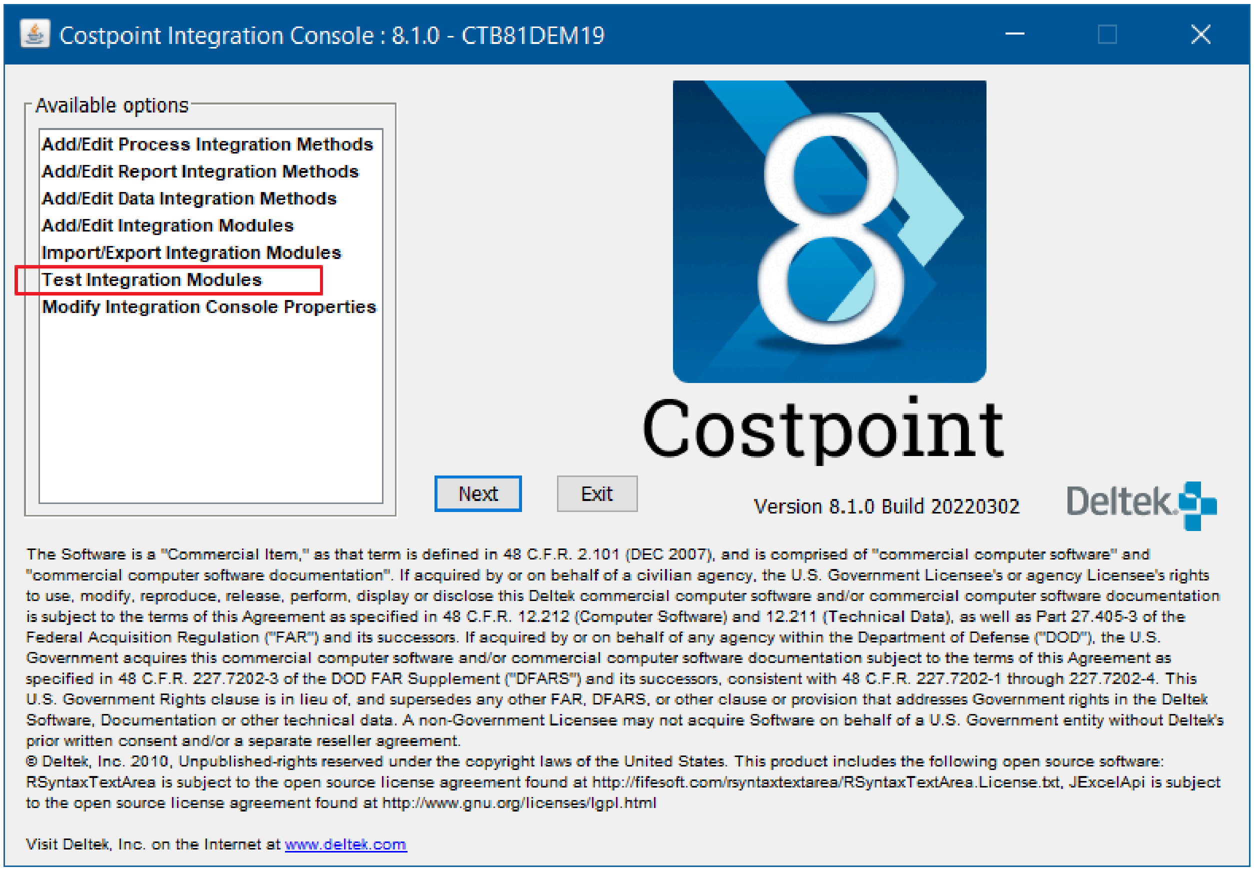
1. Select the integration module you want to test.

2. Prepare for testing by specifying testing parameters.

3. Run the test.

These steps are described in the sections that follow.

To start the testing process, select **Test Integration Modules (Web Services/EJB)** from the main Integration Console screen.


Costpoint Integration Console : 8.1.0 - CTB81DEM19
Available options
Add/Edit Process Integration Methods
Add/Edit Report Integration Methods
Add/Edit Data Integration Methods
Add/Edit Integration Modules
Import/Export Integration Modules
Test Integration Modules
Modify Integration Console Properties
8
Costpoint
Next
Exit
Version 8.1.0 Build 20220302
Deltek

The Software is a "Commercial Item," as that term is defined in 48 C.F.R. 2.101 (DEC 2007), and is comprised of "commercial computer software" and "commercial computer software documentation". If acquired by or on behalf of a civilian agency, the U.S. Government Licensee's or agency Licensee's rights to use, modify, reproduce, release, perform, display or disclose this Deltek commercial computer software and/or commercial computer software documentation is subject to the terms of this Agreement as specified in 48 C.F.R. 12.212 (Computer Software) and 12.211 (Technical Data), as well as Part 27.405-3 of the Federal Acquisition Regulation ("FAR") and its successors. If acquired by or on behalf of any agency within the Department of Defense ("DOD"), the U.S. Government acquires this commercial computer software and/or commercial computer software documentation subject to the terms of this Agreement as specified in 48 C.F.R. 227.7202-3 of the DOD FAR Supplement ("DFARS") and its successors, consistent with 48 C.F.R. 227.7202-1 through 227.7202-4. This U.S. Government Rights clause is in lieu of, and supersedes any other FAR, DFARS, or other clause or provision that addresses Government rights in the Deltek Software, Documentation or other technical data. A non-Government Licensee may not acquire Software on behalf of a U.S. Government entity without Deltek's prior written consent and/or a separate reseller agreement.

© Deltek, Inc. 2010, Unpublished-rights reserved under the copyright laws of the United States. This product includes the following open source software: RSyntaxTextArea is subject to the open source license agreement found at http://fifesoft.com/rsyntaxtextarea/RSyntaxTextArea.License.txt, JExcelApi is subject to the open source license agreement found at http://www.gnu.org/licenses/lgpl.html

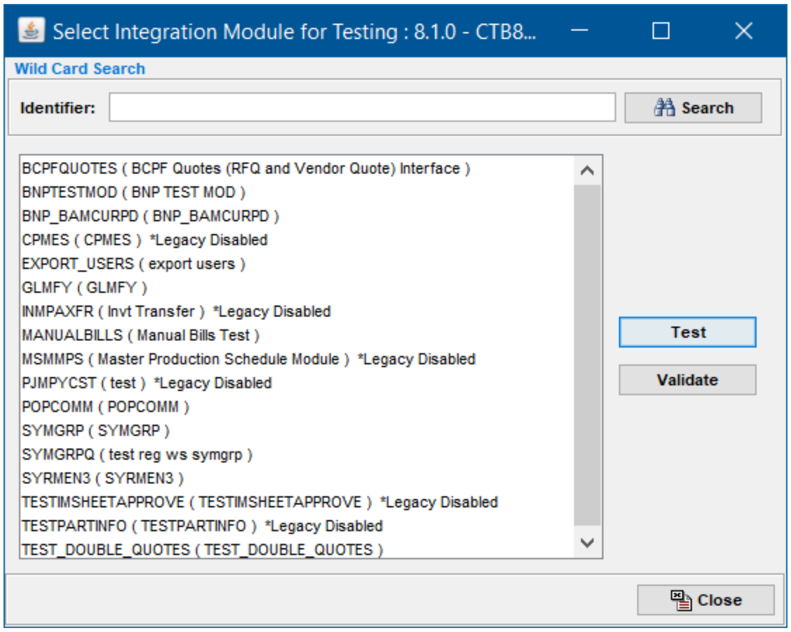
Visit Deltek, Inc. on the Internet at www.deltek.com

## Select an Integration Module for Testing

To select an integration module for testing:

1. Select **Test Integration Modules (Web Services/EJB)** from the main Integration Console screen, and click **Next**.

   The **Select Integration Module for Testing** dialog box displays.



BCPFQUOTES ( BCPF Quotes (RFQ and Vendor Quote) Interface )
BNPTESTMOD ( BNP TEST MOD )
BNP_BAMCURPD ( BNP_BAMCURPD )
CPMES ( CPMES ) *Legacy Disabled
EXPORT_USERS ( export users )
GLMFY ( GLMFY )
INMPAXFR ( Invt Transfer ) *Legacy Disabled
MANUALBILLS ( Manual Bills Test )
MSMMPS ( Master Production Schedule Module ) *Legacy Disabled
PJMPYCST ( test ) *Legacy Disabled
POPCOMM ( POPCOMM )
SYMGRP ( SYMGRP )
SYMGRPQ ( test reg ws symgrp )
SYRMEN3 ( SYRMEN3 )
TESTIMSHEETAPPROVE ( TESTIMSHEETAPPROVE ) *Legacy Disabled
TESTPARTINFO ( TESTPARTINFO ) *Legacy Disabled
TEST_DOUBLE_QUOTES ( TEST_DOUBLE_QUOTES )

2. Use these fields to select a module for testing:

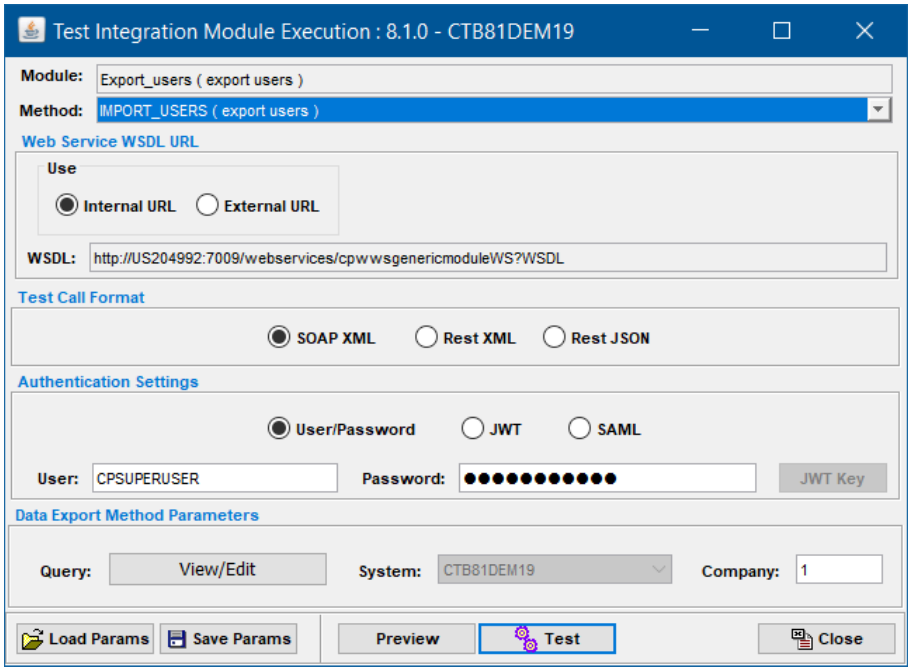
   - **Identifier**: If the module list is large, enter a filter, based on the module ID, in the **Identifier** field, and click **Search** to filter the list. The filter is performed using the LIKE condition.
   - **Test**: Select a module to test, and click this button to open the Test Module Execution dialog box. Use this dialog box to test/execute the methods in a module.
   - **Close**: Click this button to close the dialog box and return to the Costpoint Integration screen.

3. Click **Test**.

   **Note:** Deltek recommends that all testing should be done first without requiring SSL since SSL setup and troubleshooting is a manual and very time consuming process. Deltek will support SSL-related inquiries only when the SSL certificate is issued and signed by a valid CA.

## Prepare to Test an Integration Module

Follow the steps in the previous section to select the integration module you want to test. The Test Integration Module Execution dialog box displays, showing data for that integration module.



Test Integration Module Execution : 8.1.0 - CTB81DEM19

Module: Export_users ( export users )

Method: IMPORT_USERS ( export users )

Web Service WSDL URL

Use

● Internal URL ○ External URL

WSDL: http://US204992:7009/webservices/cpwwsgenericmoduleWS?WSDL

Test Call Format

● SOAP XML ○ Rest XML ○ Rest JSON

Authentication Settings

● User/Password ○ JWT ○ SAML

User: CPSUPERUSER

Password: ●●●●●●●●●●●

Data Export Method Parameters

Query:

System: CTB81DEM19

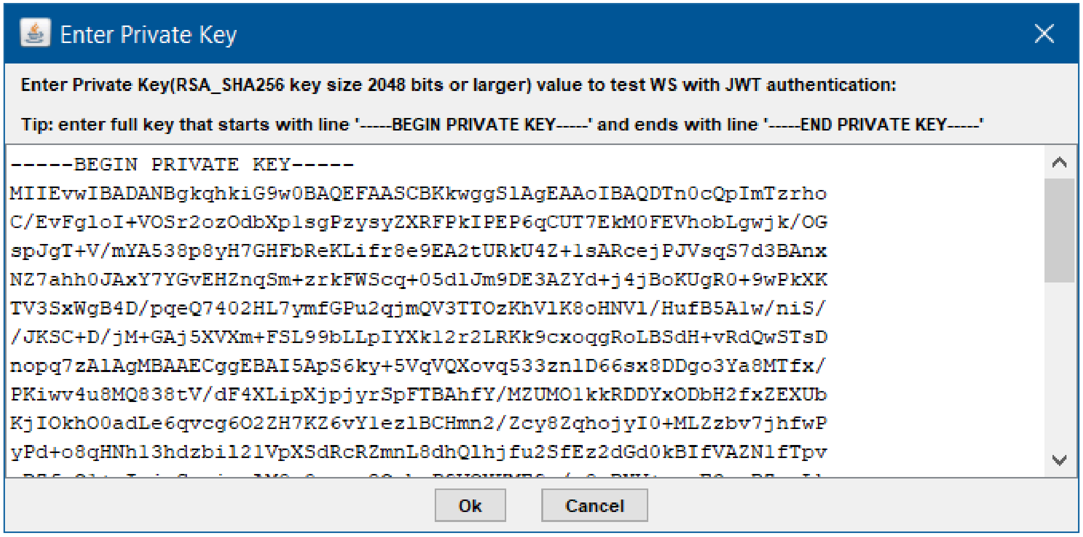
Company: 1

Review and enter data on the Test Integration Module Execution dialog box:

- Module: This field displays the module ID and description.
- Web service WSDL: This field displays the location of the Web service WSDL file.
- Internal URL: Select this option to test the Web Service against the test server IP and port.
- External URL: Select this option to test the Web Service against the external product URL (typically IIS) if it is configured.
- Method: If the module contains multiple methods, use the drop-down list in this field to select the method to test.
- Test Call Format: In Costpoint, you can invoke any Web service in any of the three supported formats: SOAP XML, Rest XML, and Rest JSON.
- In the Authentication Settings, you can select one of three available authentication options: User/Password, JWT, or SAML token.
- If you selected first User/Password option, you'll need to enter:
  - User: Enter a Costpoint user ID. This user must have the necessary application rights to execute this method. (An integration method always originates from an application.) In addition, this user must be authorized to execute an integration method.
    Attention: See the online help for the Allow Application Access via Integration Services option in the Maintain Users application for more details.
  - Password: Enter the Costpoint user password.
- If you selected JWT option, you'll need to do the following:
  - User: Enter a Costpoint user ID. This user must have the necessary application rights to execute this method. (An integration method always originates from an application.) In addition, this user must be

authorized to execute an integration method.

  - **JWT Key**: Click the **JWT Key** button and enter a private Key value in the opened dialog box. Please enter a full key value that starts with line **---BEGIN PRIVATE KEY---** and ends with **---END PRIVATE KEY---** as dialog box instructs you to do.

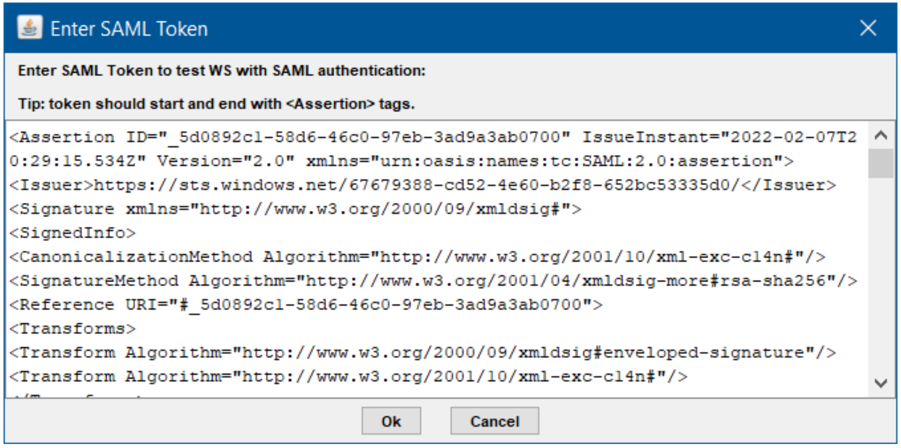


**Enter Private Key(RSA_SHA256 key size 2048 bits or larger) value to test WS with JWT authentication:**

**Tip: enter full key that starts with line '-----BEGIN PRIVATE KEY-----' and ends with line '-----END PRIVATE KEY-----'**

```
-----BEGIN PRIVATE KEY-----
MIIEvwIBADANBgkqhkiG9w0BAQEFAASCBKkwggSlAgEAAoIBAQDTn0cQpImTzrho
C/EvFgloI+VOSr2ozOdbXplsgPzysyZXRFPkIPEP6qCUT7EkM0FEVhobLgwjk/OG
spJgT+V/mYA538p8yH7GHFbReKLifr8e9EA2tURkU4Z+lsARcejPJVsqS7d3BAnx
NZ7ahh0JAxY7YGvEHZnqSm+zrkFWScq+05dlJm9DE3AZYd+j4jBoKUgR0+9wPkXK
TV3SxWgB4D/pqeQ7402HL7ymfGPu2qjmQV3TTOzKhVlK8oHNVl/HufB5Alw/niS/
/JKSC+D/jM+GAj5XVXm+FSL99bLLpIYXkl2r2LRKk9cxoqgRoLBSdH+vRdQwSTsD
nopq7zAlAgMBAAECggEBAI5ApS6ky+5VqVQXovq533znlD66sx8DDgo3Ya8MTfx/
PKiwv4u8MQ838tV/dF4XLipXjpjyrSpFTBAhfY/MZUMOlkkRDDYxODbH2fxZEXUb
KjIOkhO0adLe6qvcg6O2ZH7KZ6vYlezlBCHmn2/Zcy8ZqhojyI0+MLZzbv7jhfwP
yPd+o8qHNhl3hdzbil2lVpXSdRcRZmnL8dhQlhjfu2SfEz2dGd0kBIfVAZNlfTpv
```

- If you selected **SAML** option, you'll need to do the following:
  - **SAML Token**: Click the **SAML Token** button and enter an obtained SAML Token value in the opened dialog box. Please enter a value that starts and ends with **<Assertion>** tags as dialog box instructs you to do.

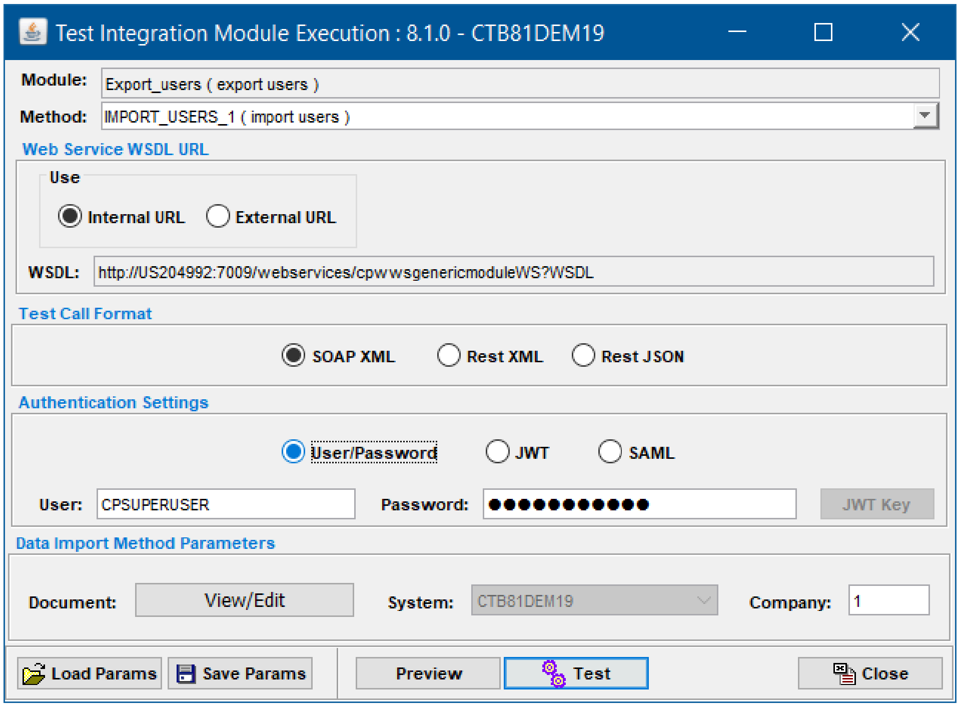


**Enter SAML Token to test WS with SAML authentication:**

**Tip: token should start and end with <Assertion> tags.**

```
<Assertion ID="_5d0892c1-58d6-46c0-97eb-3ad9a3ab0700" IssueInstant="2022-02-07T2
0:29:15.534Z" Version="2.0" xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
<Issuer>https://sts.windows.net/67679388-cd52-4e60-b2f8-652bc53335d0/</Issuer>
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
<SignedInfo>
<CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
<SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
<Reference URI="#_5d0892c1-58d6-46c0-97eb-3ad9a3ab0700">
<Transforms>
<Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
<Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
```

- In the **Method Parameters** section, you also need to specify the method data parameters related to the type of the method you are testing. For example, if you are testing the export data method, you'll need to provide a **Query** for the data you would like product to return. But if you are testing the import data method, you'll need to click **Document: View/Edit** button and provide data.



Test Integration Module Execution : 8.1.0 - CTB81DEM19

Module: Export_users ( export users )

Method: IMPORT_USERS_1 ( import users )

Web Service WSDL URL

Use: ● Internal URL ○ External URL

WSDL: http://US204992:7009/webservices/cpwwsgenericmoduleWS?WSDL

Test Call Format: ● SOAP XML ○ Rest XML ○ Rest JSON

Authentication Settings: ● User/Password ○ JWT ○ SAML

User: CPSUPERUSER

Password: ●●●●●●●●●●●

Data Import Method Parameters

System: CTB81DEM19

Company: 1

- **Query/Document**: Click the **View/Edit** button to display the Method Parameter Editor dialog box described under Specify the Query Parameters Used to Filter Integration Data.
- **System**: This field displays the system name that this method will be executed against.
- **Company**: Enter the company ID that you want to use.

## Save and Reuse Testing Parameters

You can save and reuse any set of testing parameters for future use.

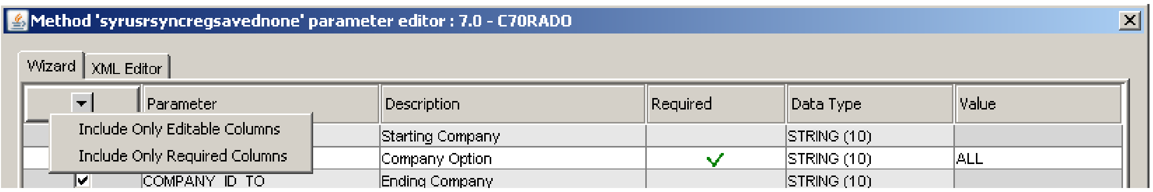
- **Save Params**: Click this button to save all testing parameters (except the password) on the Test Integration Module Execution dialog box for subsequent execution or re-execution. The data is saved locally in the user's home folder on the workstation that runs the console. For Windows workstations, the data is saved at C:\Documents and Settings\Windows User Name>.
- **Load Params**: Click this button to load a previously saved configuration.

## Preview - Testing Parameters

- **Preview**: Click this button to preview the complete SOAP packet/JSON request that will be submitted to the server. The **View/Edit** screen only shows a data portion of the message that client needs to submit to Costpoint. (This subject is covered in detail in the next section.) In the **Preview** screen, the complete message including security parameters, authentication parameters (such as user id, company, and system or JWT or SAML token), and the data parameters are displayed. This information will help you better understand how your XML or JSON request to the product should look like.

# Specify the Query Parameters Used to Filter Integration Data

The Integration Console lets you set up query filters to narrow down the pool of data on which the method will act. To set the query filters, click the **View/Edit** button on the Test Integration Module Execution dialog box to display the Method Parameter Editor screen.


Method 'syrusrsyncregsavednone' parameter editor : 7.0 - C70RADO
Wizard
XML Editor
Include Only Editable Columns
Include Only Required Columns

| | Parameter | Description | Required | Data Type | Value |
|---|---|---|---|---|---|
| | | Starting Company | | STRING (10) | |
| | | Company Option | ✓ | STRING (10) | ALL |
| ☑ | COMPANY ID TO | Ending Company | | STRING (10) | |

The information you see on the screen depends on the method type that you are testing:

- Report and process methods
  - Without saved parameters: All of the parameters to run the report or process must be supplied through the Web service client.
  - With saved parameters: The method uses a set of parameters saved previously (via either the Costpoint UI or a Data Import Integration method).
- Data export method
- Data import method

### Report or Process Method Without Saved Parameters

For this method, you can edit parameters using the Wizard or XML Editor tab on the Method Parameter Editor screen:

- The Wizard tab displays the parameters as a table that you can edit.
- The XML Editor tab displays the actual XML code and lets you edit the code.

#### Use the Wizard Tab

To use the Wizard tab:

1. Identify the parameters that you want to pass.

   For each of these parameters, you must enter a specific value in the **Value** column. If you do not pass a parameter, the system will set this parameter to a default value (if it has any) or to the null value.

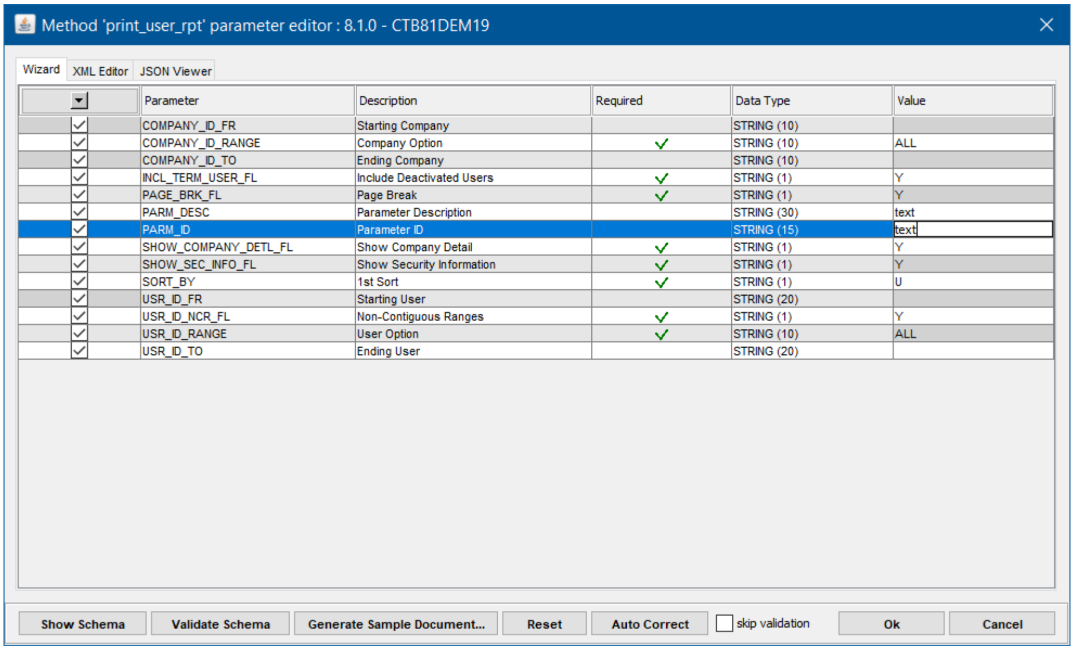
2. To make the column selection process more efficient, click the down arrow at the top left of the table,

which displays the following options:

- **Include Only Editable Columns**: Select this option to include only editable columns.
- **Include Only Required Columns**: Select this option to include only required columns.

3. Enter values for the parameters.

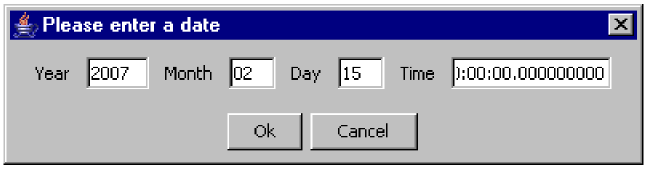
   For convenience, the console provides a drop-down list for the **Value** field whenever possible. A drop-down is provided if, in the Costpoint UI, a field is of one of the following types: combo box, range control combo box, list box, check box, or radio button.



Method 'print_user_rpt' parameter editor : 8.1.0 - CTB81DEM19

| | Parameter | Description | Required | Data Type | Value |
|---|---|---|---|---|---|
| ☑ | COMPANY_ID_FR | Starting Company | | STRING (10) | |
| ☑ | COMPANY_ID_RANGE | Company Option | ✓ | STRING (10) | ALL |
| ☑ | COMPANY_ID_TO | Ending Company | | STRING (10) | |
| ☑ | INCL_TERM_USER_FL | Include Deactivated Users | ✓ | STRING (1) | Y |
| ☑ | PAGE_BRK_FL | Page Break | ✓ | STRING (1) | Y |
| ☑ | PARM_DESC | Parameter Description | | STRING (30) | text |
| ☑ | PARM_ID | Parameter ID | | STRING (15) | text |
| ☑ | SHOW_COMPANY_DETL_FL | Show Company Detail | ✓ | STRING (1) | Y |
| ☑ | SHOW_SEC_INFO_FL | Show Security Information | ✓ | STRING (1) | Y |
| ☑ | SORT_BY | 1st Sort | ✓ | STRING (1) | U |
| ☑ | USR_ID_FR | Starting User | | STRING (20) | |
| ☑ | USR_ID_NCR_FL | Non-Contiguous Ranges | ✓ | STRING (1) | Y |
| ☑ | USR_ID_RANGE | User Option | ✓ | STRING (10) | ALL |
| ☑ | USR_ID_TO | Ending User | | STRING (20) | |

**Note:** You must use a specific format to enter dates as values, because the console uses the Java.util.Calendar, which accepts dates in the JDBC date format of yyyy-mm-dd hh:mm:ss.fffffffff. To make it easier to enter dates in this format, the console displays the following dialog box.



Please enter a date

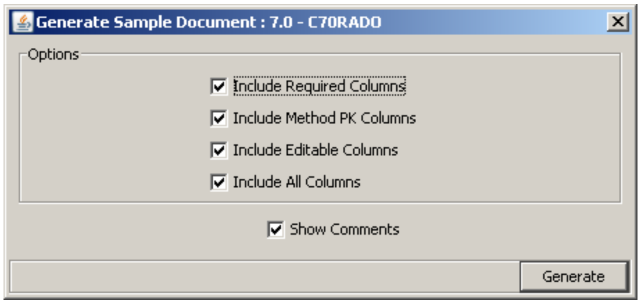
Year 2007 Month 02 Day 15 Time ):00:00.000000000

4. (Optional) Click **Show Schema** to see the method XML schema.

5. Click **Validate Schema** to test the method XML schema using sample data.

   The sample data gets transformed into an XML document before validation.

6. (Optional) Click **Generate Sample Document** to generate a sample input document.



Generate Sample Document : 7.0 - C70RADO

Options

☑ Include Required Columns

☑ Include Method PK Columns

☑ Include Editable Columns

☑ Include All Columns

☑ Show Comments

   The Generate Sample Document screen lets you select from among several options, which the Integration Console will remember until you restart the console:

   - **Include Required Columns**: Select this option to generate sample data for the required columns.
   - **Include Method PK Columns**: Select this option to generate sample data for the method PK columns.
   - **Include Editable Columns**: Select this option to generate sample data for the editable columns.
   - **Include All Columns**: Select this option to generate sample data for all the columns.
   - **Show Comments**: Select this option to include XML comments in the generated XML. The comments will be visible only when you are in the XML Editor mode.

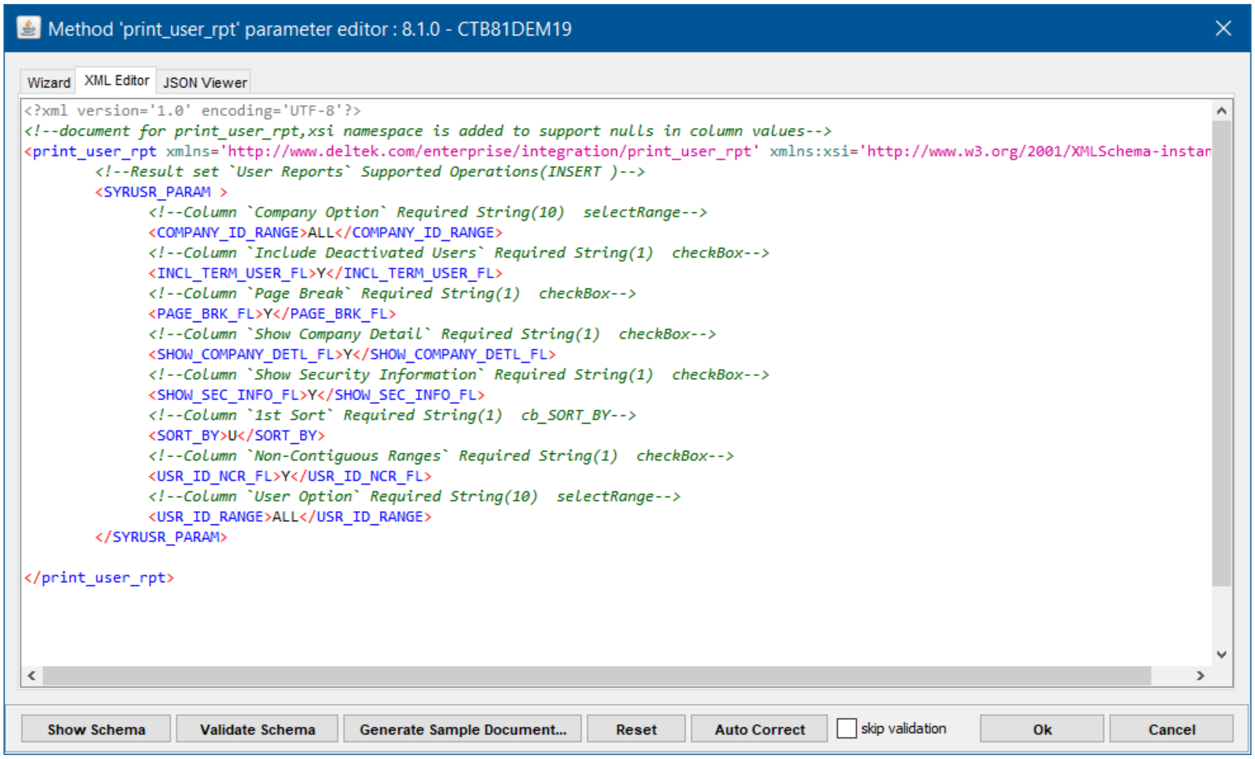
7. (Optional) Click **Generate** to generate the sample document.

8. Take one of the following actions:

   - Click **Ok** to save the generated query condition into the XML Editor tab. The query condition will be translated into an XML document.
   - Click **Reset** to reset all the changes since you opened the editor.
   - Click **Cancel** to discard all the changes and close the dialog box.

### Use the XML Editor Tab

The XML Editor tab allows you to modify the XML document directly.

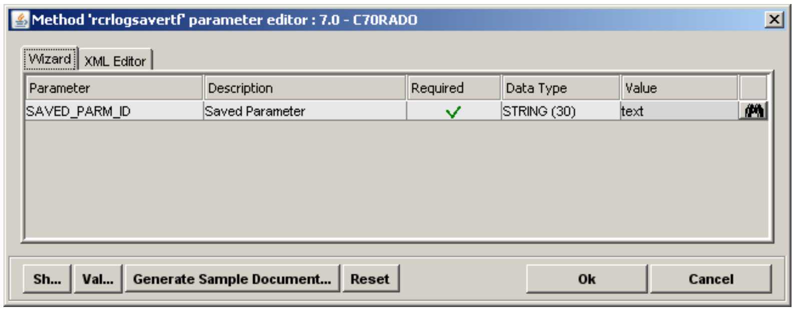


```
<?xml version='1.0' encoding='UTF-8'?>
<!--document for print_user_rpt,xsi namespace is added to support nulls in column values-->
<print_user_rpt xmlns='http://www.deltek.com/enterprise/integration/print_user_rpt' xmlns:xsi='http://www.w3.org/2001/XMLSchema-instar
        <!--Result set `User Reports` Supported Operations(INSERT )-->
        <SYRUSR_PARAM >
              <!--Column `Company Option` Required String(10)  selectRange-->
              <COMPANY_ID_RANGE>ALL</COMPANY_ID_RANGE>
              <!--Column `Include Deactivated Users` Required String(1)  checkBox-->
              <INCL_TERM_USER_FL>Y</INCL_TERM_USER_FL>
              <!--Column `Page Break` Required String(1)  checkBox-->
              <PAGE_BRK_FL>Y</PAGE_BRK_FL>
              <!--Column `Show Company Detail` Required String(1)  checkBox-->
              <SHOW_COMPANY_DETL_FL>Y</SHOW_COMPANY_DETL_FL>
              <!--Column `Show Security Information` Required String(1)  checkBox-->
              <SHOW_SEC_INFO_FL>Y</SHOW_SEC_INFO_FL>
              <!--Column `1st Sort` Required String(1)  cb_SORT_BY-->
              <SORT_BY>U</SORT_BY>
              <!--Column `Non-Contiguous Ranges` Required String(1)  checkBox-->
              <USR_ID_NCR_FL>Y</USR_ID_NCR_FL>
              <!--Column `User Option` Required String(10)  selectRange-->
              <USR_ID_RANGE>ALL</USR_ID_RANGE>
        </SYRUSR_PARAM>

</print_user_rpt>
```

You can also see the JSON corresponding JSON request structure on the JSON Viewer tab.

## Report or Process Method With Saved Parameters

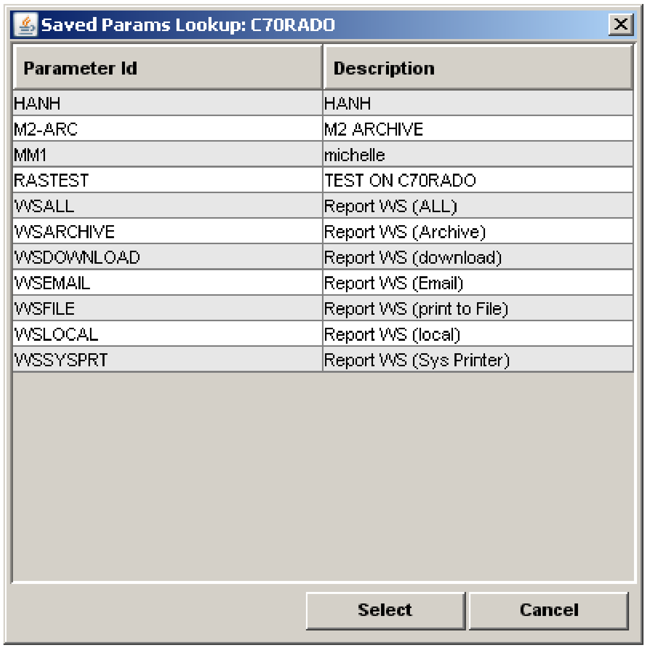


Method 'rcrlogsavertf' parameter editor : 7.0 - C70RADO
Wizard
XML Editor
Sh...
Val...
Generate Sample Document...
Reset
Ok
Cancel

| Parameter | Description | Required | Data Type | Value |
|---|---|---|---|---|
| SAVED_PARM_ID | Saved Parameter | ✓ | STRING (30) | text |

If you are testing a report or process method that uses saved parameters, a single parameter, named **SAVED_PARM_ID,** displays in the Method Parameter Editor screen. In the **Value** column, enter the parameter ID to be executed. The parameter must have been saved previously (via either the Costpoint UI or a Data Import Integration method).

Click the binoculars icon to display previously saved parameters. Select a parameter ID from this lookup.

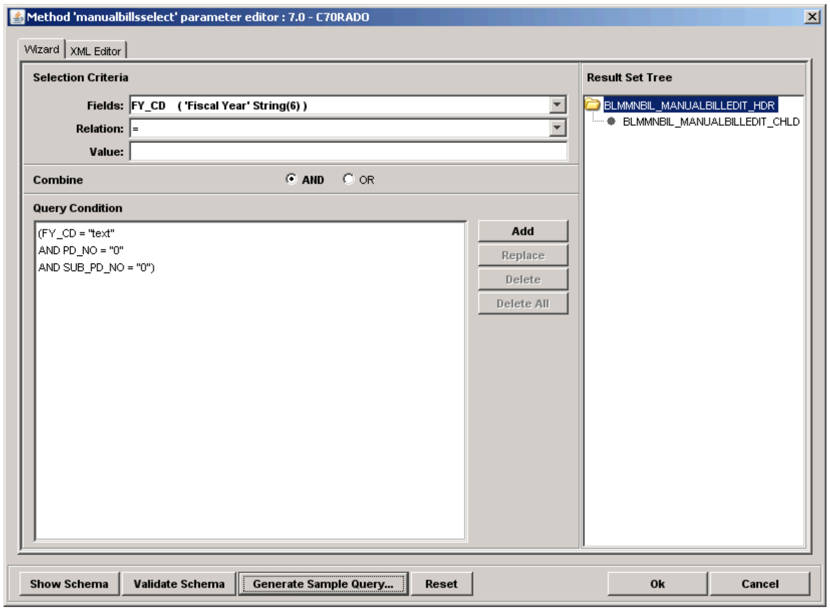


Saved Params Lookup: C70RADO

| Parameter Id | Description |
|---|---|
| HANH | HANH |
| M2-ARC | M2 ARCHIVE |
| MM1 | michelle |
| RASTEST | TEST ON C70RADO |
| WSALL | Report WS (ALL) |
| WSARCHIVE | Report WS (Archive) |
| WSDOWNLOAD | Report WS (download) |
| WSEMAIL | Report WS (Email) |
| WSFILE | Report WS (print to File) |
| WSLOCAL | Report WS (local) |
| WSSYSPRT | Report WS (Sys Printer) |

## Data Export Method

If you are testing a data export method, you must construct a query string based on the columns available for export.



Method 'manualbillsselect' parameter editor : 7.0 - C70RADO

Selection Criteria

Fields: FY_CD ( 'Fiscal Year' String(6) )

Relation: =

Value:

Combine ● AND ○ OR

Query Condition

```
(FY_CD = "text"
AND PD_NO = "0"
AND SUB_PD_NO = "0")
```

Result Set Tree

- BLMMNBIL_MANUALBILLEDIT_HDR
  - BLMMNBIL_MANUALBILLEDIT_CHLD

You can build the query using the Wizard or XML Editor tab on the Method Parameter Editor screen:

- The Wizard tab provides a series of query-building fields that let you create the query.
- The XML Editor tab displays the query condition string and lets you directly edit the code. The changes you make on this tab are synchronized back to the Wizard tab, so you can use both methods in the same session.

## Use the Wizard Tab

### To build the query using the Wizard tab:

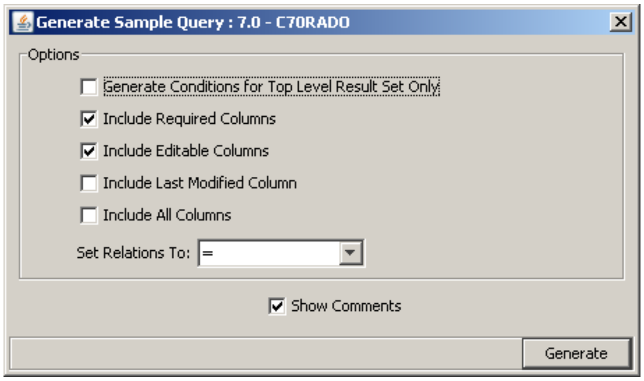
1. (Optional) Click **Show Schema** to see the method XML schema.

2. (Optional) Click **Validate Schema** to test the method XML schema using sample data.

   The sample data gets transformed into an XML document before validation.

3. Click **Generate Sample Query** to generate a sample query.

   The Generate Sample Query dialog box lets you select from among several options, which the Integration Console will remember until you restart the console:



Generate Sample Query : 7.0 - C70RADO

Options

☐ Generate Conditions for Top Level Result Set Only
☑ Include Required Columns
☑ Include Editable Columns
☐ Include Last Modified Column
☐ Include All Columns

Set Relations To: =

☑ Show Comments

- **Generate Conditions for Top Level Result Set Only**: If you have several result sets, select this option to generate query conditions only for the top level result set. Leave this option cleared to generate query conditions for all result sets.
- **Include Required Columns**: Select this option to generate sample query conditions on required columns.
- **Include Editable Columns**: Select this option to generate sample query conditions on the editable columns.
- **Include All Columns**: Select this option to generate sample query conditions on all columns.
- **Include Last Modified Column**: Select this option to generate a sample query condition on the Last Modified Pseudo Column if a particular result set has one.
- **Set Relations To**: Any condition on a result set column has several parts: column, relation operator, and value. Several relation operators (for example, **is null**) do not accept values. Select the **Set Relations To** option to choose which relation operator to use in generating sample query conditions. If you select **random**, the console will use a random algorithm when generating a relation operator. You can also select

one of the common operators for all data type relation operators (for example, = (equal)).

- **Show Comments**: Select this option to include XML comments in the generated XML. The comments will be visible only when you are in the XML Editor mode.

4. Click **Generate** to generate the sample query.

5. If your export method involves more than one result set, select the result set from the **Result Set Tree**.

   Then build the **Query Condition** for the result set using the **Selection Criteria** fields:

   - **Fields**: From the drop-down list, select the fields available in the export method.
   - **Relation**: From the drop-down list, select the relation condition.
   - **Value**: Enter the corresponding value for the selected field and relation.
   - **Combine**: Use the **AND** and **OR** options to add additional conditions.
   - **Add**: Click this button to add the condition to the **Query Condition** box.
   - **Replace**: Highlight an existing condition in the **Query Condition** box and click this button to replace it with the new condition.
   - **Delete**: Highlight an existing condition in the **Query Condition** box and click this button to delete it.
   - **Delete All**: Click this button to clear all conditions in the **Query Condition** box.

6. Take one of the following actions:

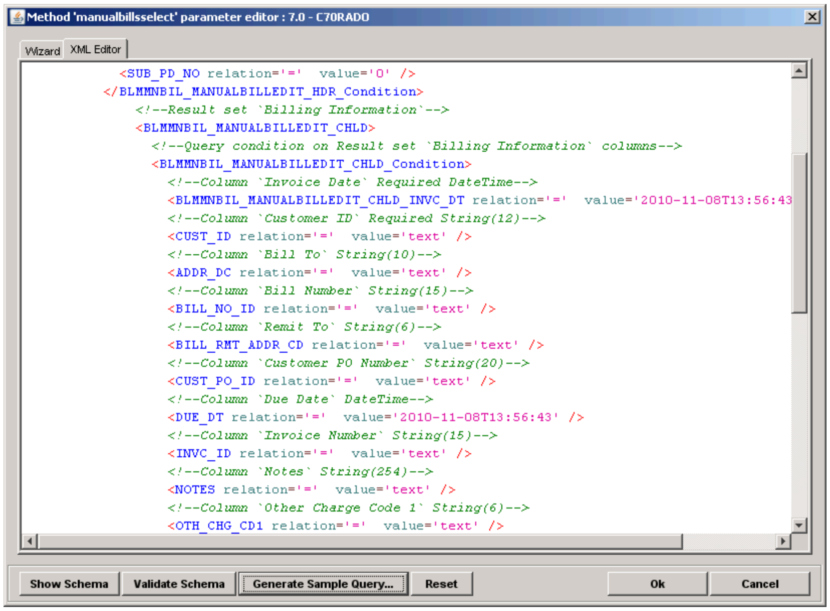
   - Click **Ok** to save the generated query condition into the **Query** field on the Test Module dialog box. The query condition will be translated into an XML document.
   - Click **Cancel** to discard all the changes and close dialog box.

### Use the XML Editor Tab

**To review or edit the query using the XML Editor tab:**

1. Click **View/Edit** on the Test Integration Module Execution dialog box.

2. Click the XML Editor tab, which displays the query condition string.



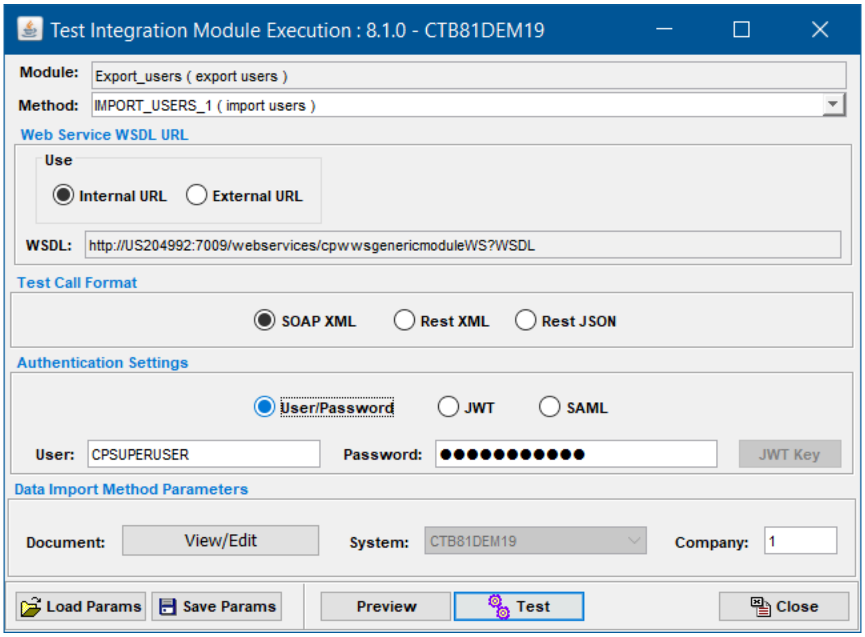
```
    <SUB_PD_NO relation='='  value='0' />
  </BLMMNBIL_MANUALBILLEDIT_HDR_Condition>
      <!--Result set `Billing Information`-->
      <BLMMNBIL_MANUALBILLEDIT_CHLD>
        <!--Query condition on Result set `Billing Information` columns-->
        <BLMMNBIL_MANUALBILLEDIT_CHLD_Condition>
          <!--Column `Invoice Date` Required DateTime-->
          <BLMMNBIL_MANUALBILLEDIT_CHLD_INVC_DT relation='='  value='2010-11-08T13:56:43
          <!--Column `Customer ID` Required String(12)-->
          <CUST_ID relation='='  value='text' />
          <!--Column `Bill To` String(10)-->
          <ADDR_DC relation='='  value='text' />
          <!--Column `Bill Number` String(15)-->
          <BILL_NO_ID relation='='  value='text' />
          <!--Column `Remit To` String(6)-->
          <BILL_RMT_ADDR_CD relation='='  value='text' />
          <!--Column `Customer PO Number` String(20)-->
          <CUST_PO_ID relation='='  value='text' />
          <!--Column `Due Date` DateTime-->
          <DUE_DT relation='='  value='2010-11-08T13:56:43' />
          <!--Column `Invoice Number` String(15)-->
          <INVC_ID relation='='  value='text' />
          <!--Column `Notes` String(254)-->
          <NOTES relation='='  value='text' />
          <!--Column `Other Charge Code 1` String(6)-->
          <OTH_CHG_CD1 relation='='  value='text' />
```

3. Review and, if necessary, edit the query.

   Any changes you make on this tab are synchronized back to the Wizard tab, so you can use both methods in the same session.

### Data Import Method

If you are testing a data import method, you must specify the data that will be imported. You do this by directly editing the XML document that contains the data for your import method.



Test Integration Module Execution : 8.1.0 - CTB81DEM19

Module: Export_users ( export users )

Method: IMPORT_USERS_1 ( import users )

Web Service WSDL URL

Use

● Internal URL ○ External URL

WSDL: http://US204992:7009/webservices/cpwwsgenericmoduleWS?WSDL

Test Call Format

● SOAP XML ○ Rest XML ○ Rest JSON

Authentication Settings

● User/Password ○ JWT ○ SAML

User: CPSUPERUSER Password: ●●●●●●●●●●●

Data Import Method Parameters

Document: System: CTB81DEM19 Company: 1

To specify the data that will be imported:

1. Click **View/Edit** on the Test Integration Module Execution dialog box.

2. (Optional) On the XML Editor tab, click **Generate Sample Document** to generate a sample XML document to edit.

   You can also see the corresponding JSON request in a **JSON Viewer** tab

   **Attention:** See step 6 for more information about generating sample documents.

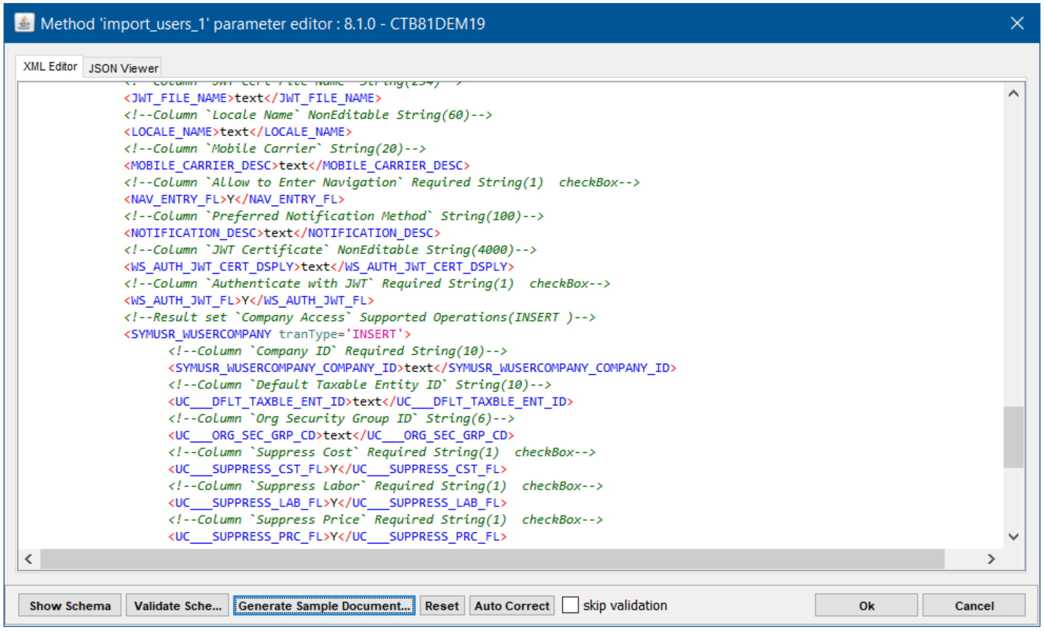
3. Edit the XML document that contains the data for your import method.

   You can use four operations with the import method: **INSERT**, **UPDATE**, **DELETE**, **MERGE**, and **SELECT**. Each result set that you specify has a tranType attribute. Set this attribute to the operation that you want to perform in this test. If you omit this attribute, the console assumes you are specifying an **INSERT** operation.

   For each column included in the import method, include the tag and the value to be used:

   - **INSERT**: All required columns (as indicated in the XML schema) should be present and their values included in the document. A required column is indicated with the attribute **nillable='false'**.
   - **UPDATE** operation: Only columns that need to be updated should be included in the document. If a column needs to be updated to a blank value, include the tag in the document but omit the value. For example, enter **<NOTES></NOTES>**.

- MERGE operation: Use this to INSERT new records or UPDATE existing records, depending on whether or not a record already exist.
- DELETE operation: All columns specified as primary keys (PK) when you created the import method must be present and have values.
- SELECT operation: All columns specified as primary keys (PK) when you created the import method must be present and have values. You must use the SELECT operation when you want to update a child result set only. In this case, you use the SELECT operation to select a row in the parent result set, and then you use INSERT, DELETE, or UPDATE to modify a child row.



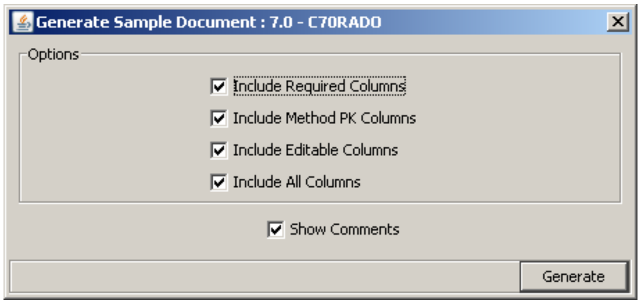
```
<!--Column `JWT Cert File Name` String(254)-->
<JWT_FILE_NAME>text</JWT_FILE_NAME>
<!--Column `Locale Name` NonEditable String(60)-->
<LOCALE_NAME>text</LOCALE_NAME>
<!--Column `Mobile Carrier` String(20)-->
<MOBILE_CARRIER_DESC>text</MOBILE_CARRIER_DESC>
<!--Column `Allow to Enter Navigation` Required String(1)  checkBox-->
<NAV_ENTRY_FL>Y</NAV_ENTRY_FL>
<!--Column `Preferred Notification Method` String(100)-->
<NOTIFICATION_DESC>text</NOTIFICATION_DESC>
<!--Column `JWT Certificate` NonEditable String(4000)-->
<WS_AUTH_JWT_CERT_DSPLY>text</WS_AUTH_JWT_CERT_DSPLY>
<!--Column `Authenticate with JWT` Required String(1)  checkBox-->
<WS_AUTH_JWT_FL>Y</WS_AUTH_JWT_FL>
<!--Result set `Company Access` Supported Operations(INSERT )-->
<SYMUSR_WUSERCOMPANY tranType='INSERT'>
      <!--Column `Company ID` Required String(10)-->
      <SYMUSR_WUSERCOMPANY_COMPANY_ID>text</SYMUSR_WUSERCOMPANY_COMPANY_ID>
      <!--Column `Default Taxable Entity ID` String(10)-->
      <UC___DFLT_TAXBLE_ENT_ID>text</UC___DFLT_TAXBLE_ENT_ID>
      <!--Column `Org Security Group ID` String(6)-->
      <UC___ORG_SEC_GRP_CD>text</UC___ORG_SEC_GRP_CD>
      <!--Column `Suppress Cost` Required String(1)  checkBox-->
      <UC___SUPPRESS_CST_FL>Y</UC___SUPPRESS_CST_FL>
      <!--Column `Suppress Labor` Required String(1)  checkBox-->
      <UC___SUPPRESS_LAB_FL>Y</UC___SUPPRESS_LAB_FL>
      <!--Column `Suppress Price` Required String(1)  checkBox-->
      <UC___SUPPRESS_PRC_FL>Y</UC___SUPPRESS_PRC_FL>
```

4. (Optional) Click **Show Schema** to see the XML schema created for this import method (with all the selections and settings) when the import method was created.

5. Click **Validate Schema** to validate your document to see if it conforms to the XML schema for this method.

6. Click **Generate Sample Document** to generate a sample input document.

   The Generate Sample Document screen lets you select from among several options, which the Integration Console will remember until you restart the console:



Generate Sample Document : 7.0 - C70RADO

Options

☑ Include Required Columns

☑ Include Method PK Columns

☑ Include Editable Columns

☑ Include All Columns

☑ Show Comments

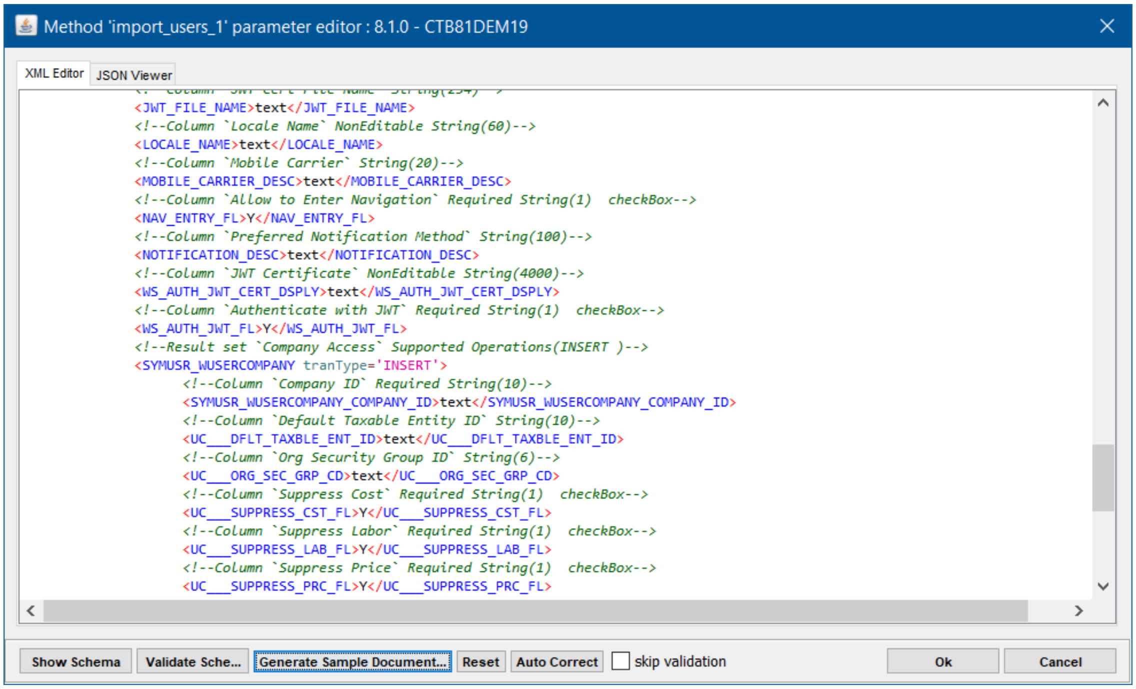
- **Include Required Columns**: Select this option to generate sample data for required columns.
- **Include Method PK Columns**: Select this option to generate sample data for method PK columns.
- **Include Editable Columns**: Select this option to generate sample data for editable columns.
- **Include All Columns**: Select this option to generate sample data for all columns.
- **Show Comments**: Select this option to include XML comments in the generated XML. The comments will be visible only when you are in the XML Editor mode.

7. Click **Generate.**

   The Integration Console generates one row for each result set that you included in the method.

If you decided to pass data as an object, the Method Parameter Editor screen displays. The options on the screen depend on the method type. Plus, there is an extra view that represents your query condition or input document as a JSON type that will be invoked by the test method if JSON is selected as a format.

The following is an XML Editor view of the data import method editor.

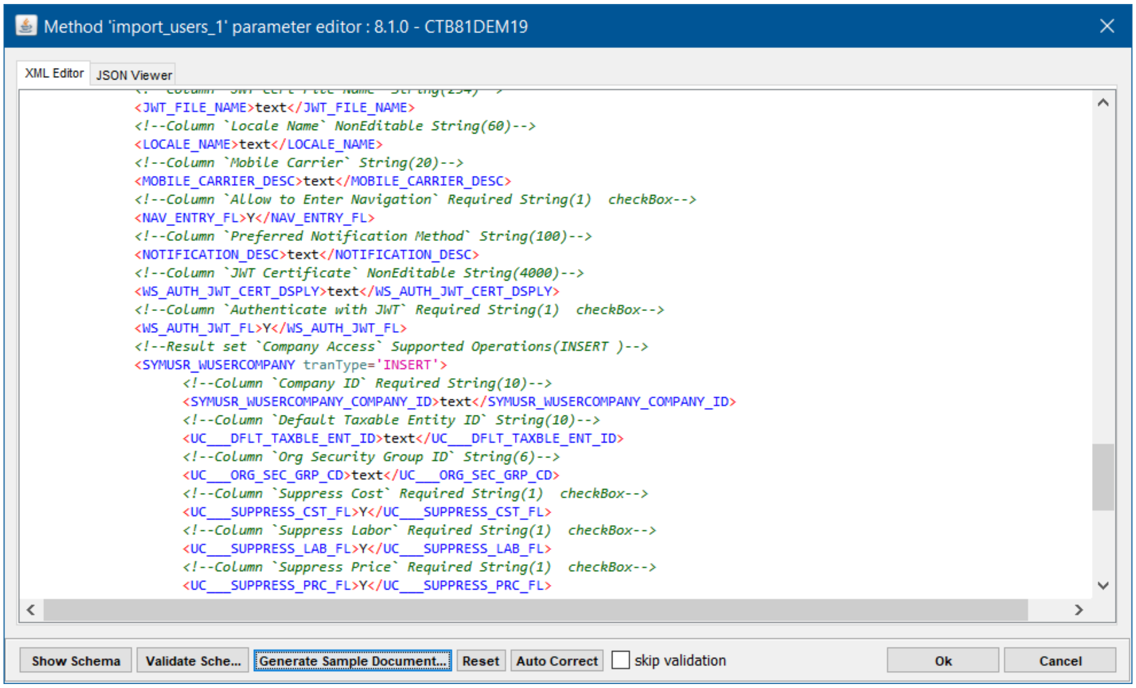


Method 'import_users_1' parameter editor : 8.1.0 - CTB81DEM19

```
<JWT_FILE_NAME>text</JWT_FILE_NAME>
<!--Column `Locale Name` NonEditable String(60)-->
<LOCALE_NAME>text</LOCALE_NAME>
<!--Column `Mobile Carrier` String(20)-->
<MOBILE_CARRIER_DESC>text</MOBILE_CARRIER_DESC>
<!--Column `Allow to Enter Navigation` Required String(1)  checkBox-->
<NAV_ENTRY_FL>Y</NAV_ENTRY_FL>
<!--Column `Preferred Notification Method` String(100)-->
<NOTIFICATION_DESC>text</NOTIFICATION_DESC>
<!--Column `JWT Certificate` NonEditable String(4000)-->
<WS_AUTH_JWT_CERT_DSPLY>text</WS_AUTH_JWT_CERT_DSPLY>
<!--Column `Authenticate with JWT` Required String(1)  checkBox-->
<WS_AUTH_JWT_FL>Y</WS_AUTH_JWT_FL>
<!--Result set `Company Access` Supported Operations(INSERT )-->
<SYMUSR_WUSERCOMPANY tranType='INSERT'>
      <!--Column `Company ID` Required String(10)-->
      <SYMUSR_WUSERCOMPANY_COMPANY_ID>text</SYMUSR_WUSERCOMPANY_COMPANY_ID>
      <!--Column `Default Taxable Entity ID` String(10)-->
      <UC___DFLT_TAXBLE_ENT_ID>text</UC___DFLT_TAXBLE_ENT_ID>
      <!--Column `Org Security Group ID` String(6)-->
      <UC___ORG_SEC_GRP_CD>text</UC___ORG_SEC_GRP_CD>
      <!--Column `Suppress Cost` Required String(1)  checkBox-->
      <UC___SUPPRESS_CST_FL>Y</UC___SUPPRESS_CST_FL>
      <!--Column `Suppress Labor` Required String(1)  checkBox-->
      <UC___SUPPRESS_LAB_FL>Y</UC___SUPPRESS_LAB_FL>
      <!--Column `Suppress Price` Required String(1)  checkBox-->
      <UC___SUPPRESS_PRC_FL>Y</UC___SUPPRESS_PRC_FL>
```

☐ skip validation

The following is a JSON view of the data import method editor.

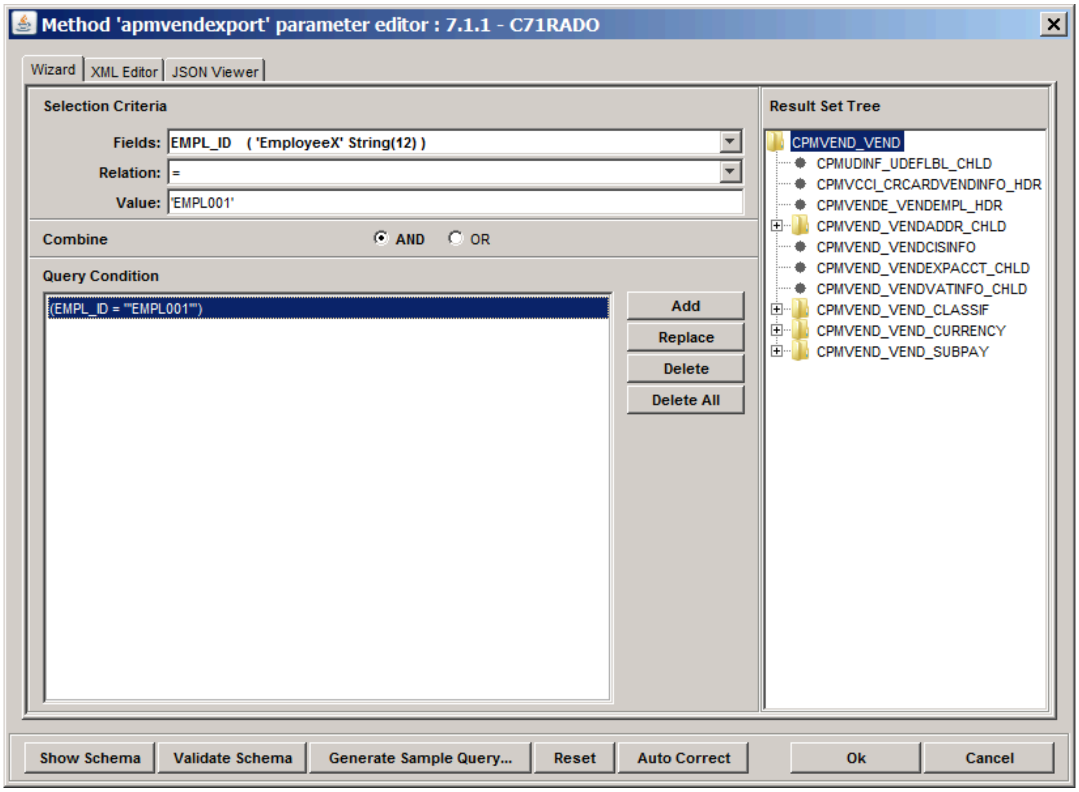


```
<JWT_FILE_NAME>text</JWT_FILE_NAME>
<!--Column `Locale Name` NonEditable String(60)-->
<LOCALE_NAME>text</LOCALE_NAME>
<!--Column `Mobile Carrier` String(20)-->
<MOBILE_CARRIER_DESC>text</MOBILE_CARRIER_DESC>
<!--Column `Allow to Enter Navigation` Required String(1)  checkBox-->
<NAV_ENTRY_FL>Y</NAV_ENTRY_FL>
<!--Column `Preferred Notification Method` String(100)-->
<NOTIFICATION_DESC>text</NOTIFICATION_DESC>
<!--Column `JWT Certificate` NonEditable String(4000)-->
<WS_AUTH_JWT_CERT_DSPLY>text</WS_AUTH_JWT_CERT_DSPLY>
<!--Column `Authenticate with JWT` Required String(1)  checkBox-->
<WS_AUTH_JWT_FL>Y</WS_AUTH_JWT_FL>
<!--Result set `Company Access` Supported Operations(INSERT )-->
<SYMUSR_WUSERCOMPANY tranType='INSERT'>
      <!--Column `Company ID` Required String(10)-->
      <SYMUSR_WUSERCOMPANY_COMPANY_ID>text</SYMUSR_WUSERCOMPANY_COMPANY_ID>
      <!--Column `Default Taxable Entity ID` String(10)-->
      <UC___DFLT_TAXBLE_ENT_ID>text</UC___DFLT_TAXBLE_ENT_ID>
      <!--Column `Org Security Group ID` String(6)-->
      <UC___ORG_SEC_GRP_CD>text</UC___ORG_SEC_GRP_CD>
      <!--Column `Suppress Cost` Required String(1)  checkBox-->
      <UC___SUPPRESS_CST_FL>Y</UC___SUPPRESS_CST_FL>
      <!--Column `Suppress Labor` Required String(1)  checkBox-->
      <UC___SUPPRESS_LAB_FL>Y</UC___SUPPRESS_LAB_FL>
      <!--Column `Suppress Price` Required String(1)  checkBox-->
      <UC___SUPPRESS_PRC_FL>Y</UC___SUPPRESS_PRC_FL>
```

The buttons at the bottom of the screen let you perform these actions:

- Show Schema: Click this button to display the data schema for your method.
- Validate Schema: Click this button to validate your Java Class by compiling it.
- Generate Sample Document: Click this button to generate a sample document.
- Ok: Click this button to close the Editor screen after validating the Java Class by compiling it and updating your query condition. If you are currently in the XML View, the document will be translated into Java Class format before validation.
- Cancel: Click this button to close the screen without updating your query condition.

In addition to the standard **SOAP XML** format, the Generic/Regular Web Services product supports invoking Web Services in **Rest XML** and **Rest JSON** formats. To learn more about these options, Deltek recommends using the View/Edit Query dialog box, which can show you how to construct XML/JSON for an entered query condition.



Method 'apmvendexport' parameter editor : 7.1.1 - C71RADO

Selection Criteria

Fields: EMPL_ID ( 'EmployeeX' String(12) )

Relation: =

Value: 'EMPL001'

Combine ● AND ○ OR

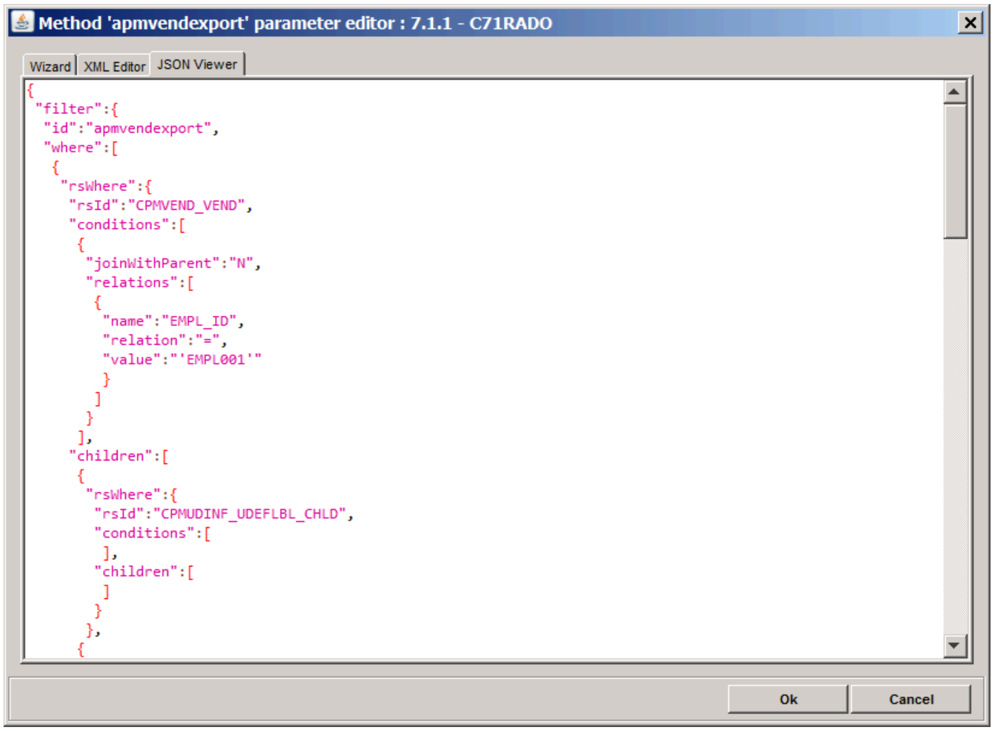
Query Condition

(EMPL_ID = "'EMPL001'")

Result Set Tree

- CPMVEND_VEND
  - CPMUDINF_UDEFLBL_CHLD
  - CPMVCCI_CRCARDVENDINFO_HDR
  - CPMVENDE_VENDEMPL_HDR
  - CPMVEND_VENDADDR_CHLD
  - CPMVEND_VENDCISINFO
  - CPMVEND_VENDEXPACCT_CHLD
  - CPMVEND_VENDVATINFO_CHLD
  - CPMVEND_VEND_CLASSIF
  - CPMVEND_VEND_CURRENCY
  - CPMVEND_VEND_SUBPAY

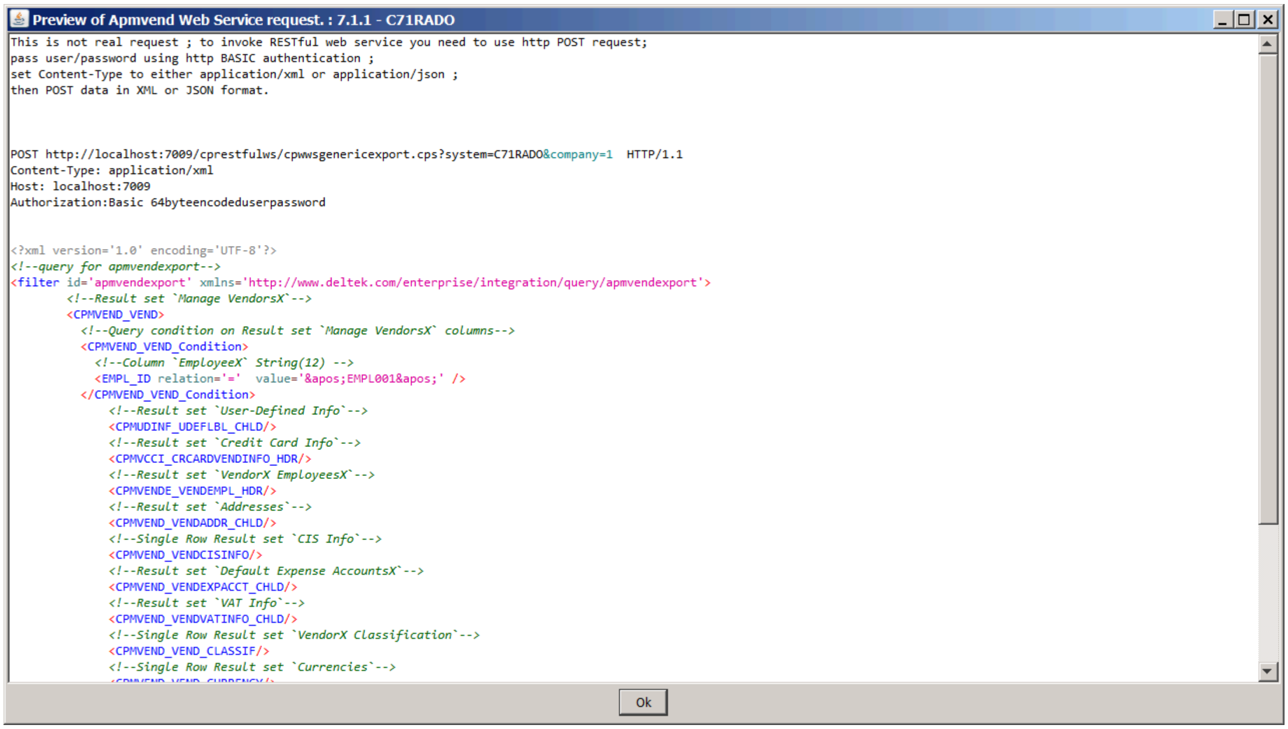
On the XML Edior/JSON Viewer tab:



Method 'apmvendexport' parameter editor : 7.1.1 - C71RADO

```
{
 "filter":{
  "id":"apmvendexport",
  "where":[
   {
    "rsWhere":{
     "rsId":"CPMVEND_VEND",
     "conditions":[
      {
       "joinWithParent":"N",
       "relations":[
        {
         "name":"EMPL_ID",
         "relation":"=",
         "value":"'EMPL001'"
         }
        ]
       }
      ],
     "children":[
      {
       "rsWhere":{
        "rsId":"CPMUDINF_UDEFLBL_CHLD",
        "conditions":[
         ],
        "children":[
         ]
        }
       },
      {
```

And use the Preview dialog box for **Rest XML**:



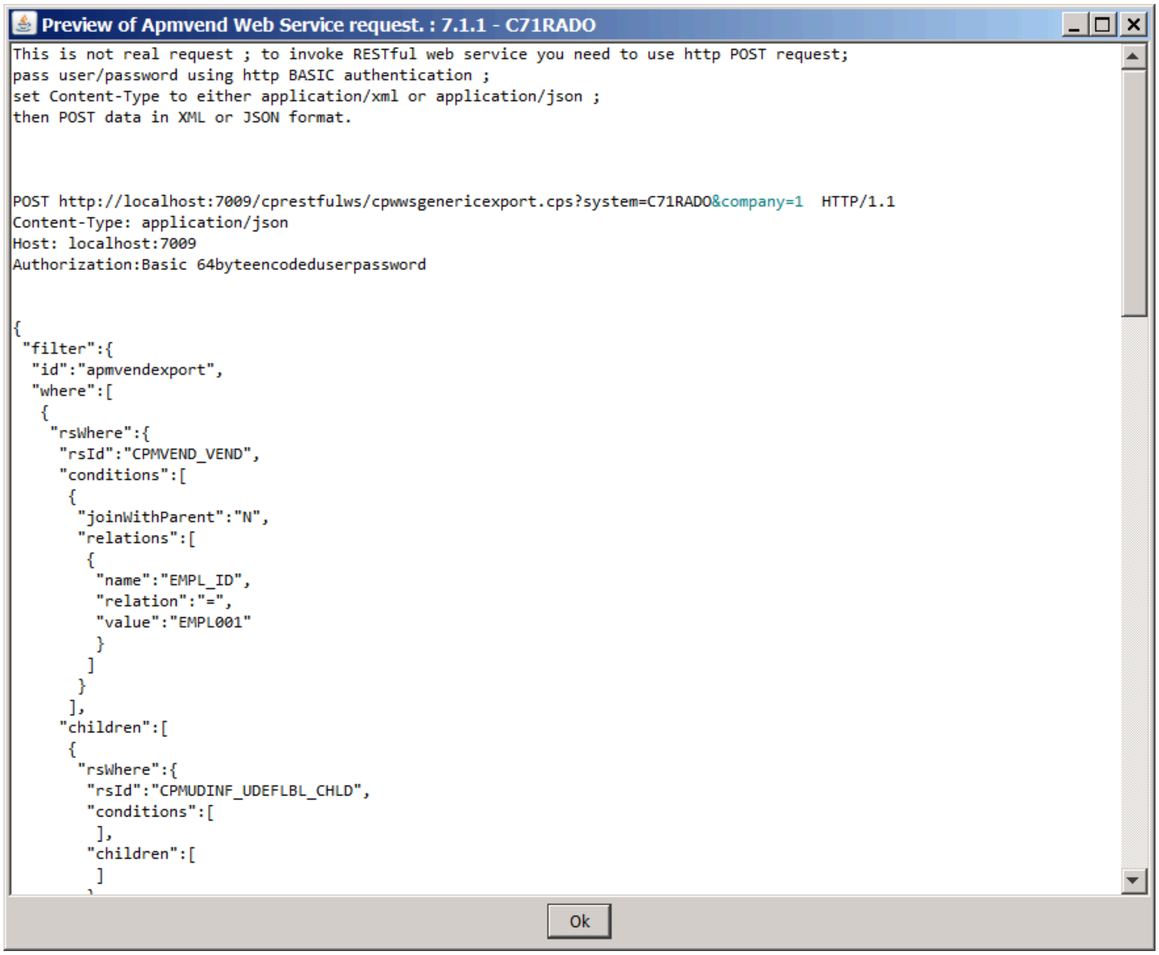
```
This is not real request ; to invoke RESTful web service you need to use http POST request;
pass user/password using http BASIC authentication ;
set Content-Type to either application/xml or application/json ;
then POST data in XML or JSON format.



POST http://localhost:7009/cprestfulws/cpwwsgenericexport.cps?system=C71RADO&company=1  HTTP/1.1
Content-Type: application/xml
Host: localhost:7009
Authorization:Basic 64byteencodeduserpassword


<?xml version='1.0' encoding='UTF-8'?>
<!--query for apmvendexport-->
<filter id='apmvendexport' xmlns='http://www.deltek.com/enterprise/integration/query/apmvendexport'>
        <!--Result set `Manage VendorsX`-->
        <CPMVEND_VEND>
          <!--Query condition on Result set `Manage VendorsX` columns-->
          <CPMVEND_VEND_Condition>
            <!--Column `EmployeeX` String(12) -->
            <EMPL_ID relation='='  value='&apos;EMPL001&apos;' />
          </CPMVEND_VEND_Condition>
              <!--Result set `User-Defined Info`-->
              <CPMUDINF_UDEFLBL_CHLD/>
              <!--Result set `Credit Card Info`-->
              <CPMVCCI_CRCARDVENDINFO_HDR/>
              <!--Result set `VendorX EmployeesX`-->
              <CPMVENDE_VENDEMPL_HDR/>
              <!--Result set `Addresses`-->
              <CPMVEND_VENDADDR_CHLD/>
              <!--Single Row Result set `CIS Info`-->
              <CPMVEND_VENDCISINFO/>
              <!--Result set `Default Expense AccountsX`-->
              <CPMVEND_VENDEXPACCT_CHLD/>
              <!--Result set `VAT Info`-->
              <CPMVEND_VENDVATINFO_CHLD/>
              <!--Single Row Result set `VendorX Classification`-->
              <CPMVEND_VEND_CLASSIF/>
              <!--Single Row Result set `Currencies`-->
              <CPMVEND_VEND_CURRENCY/>
```

Or **Rest JSON** format:



```
This is not real request ; to invoke RESTful web service you need to use http POST request;
pass user/password using http BASIC authentication ;
set Content-Type to either application/xml or application/json ;
then POST data in XML or JSON format.



POST http://localhost:7009/cprestfulws/cpwwsgenericexport.cps?system=C71RADO&company=1  HTTP/1.1
Content-Type: application/json
Host: localhost:7009
Authorization:Basic 64byteencodeduserpassword


{
 "filter":{
  "id":"apmvendexport",
  "where":[
   {
    "rsWhere":{
     "rsId":"CPMVEND_VEND",
     "conditions":[
      {
       "joinWithParent":"N",
       "relations":[
        {
         "name":"EMPL_ID",
         "relation":"=",
         "value":"EMPL001"
         }
        ]
       }
      ],
     "children":[
      {
       "rsWhere":{
        "rsId":"CPMUDINF_UDEFLBL_CHLD",
        "conditions":[
         ],
        "children":[
         ]
```

Those Preview dialog boxes contain more information on how to invoke Web Service in the desired format.

## Run the Test of an Integration Module

Click Test on the Test Integration Module Execution dialog box to start method execution. The console performs basic parameter validation checks to ensure that:

- Required parameters are not null
- Entered parameters represent valid values according to their data types (double, calendar, string).

If no errors are found, the Testing Process status screen displays, showing the progress of the execution.

For Web Services troubleshooting purposes, you can generate special Debugging Web Services log file that can help you troubleshoot Web Services invocation problems. This Option can be turned on or off in the Configuration Utility » Logging tab.

**Attention:** For more information please see *Deltek Costpoint Configuration Utility* guide.

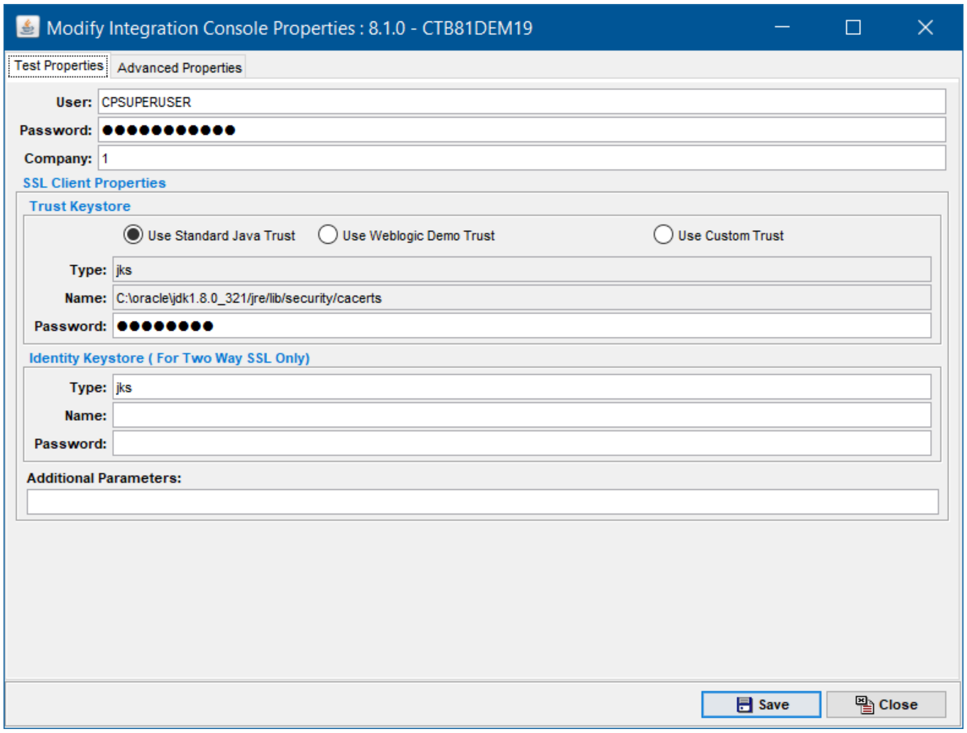
# Modify Integration Console Properties

Environment settings used by the Integration Console are stored in Costpoint configuration files. This section describes the settings that can be modified in this tool.

Select **Modify Integration Console Properties** from the main Integration Console screen, and click **Next**.

### Test Properties Tab

Use this tab to modify default settings used in the test process.



User: CPSUPERUSER
Password: ●●●●●●●●●●●●
Company: 1

SSL Client Properties

Trust Keystore

● Use Standard Java Trust ○ Use Weblogic Demo Trust ○ Use Custom Trust

Type: jks
Name: C:\oracle\jdk1.8.0_321/jre/lib/security/cacerts
Password: ●●●●●●●●

Identity Keystore ( For Two Way SSL Only)

Type: jks
Name:
Password:

Additional Parameters:

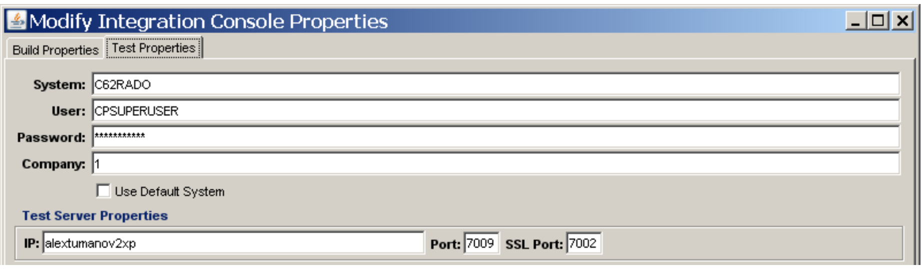
The fields on the screen are:

- **User**: Enter the default user name. This is the user ID that is used to connect to Costpoint Server to run the test. This user must be authorized (in the Costpoint Manage Users application) to access Costpoint via Integration Services.
- **Password**: Enter the password for the default user or leave blank to enter the password on the testing screen.
- **Company**: Enter the default company name. The Integration Console needs this information to execute modules.

- **SSL Client Properties**: If SSL is the chosen protocol, these settings are used by the console during module deployment and execution.
    - **Trust Keystore**: Specify the SSL properties that are used in both One Way and Two Way SSL protocols. Supported Trust Keystore options are:
        - Use Standard Java Trust
        - Use WebLogic Demo Trust
        - Use Custom Trust
          Depending on the chosen Trust Keystore, some additional fields in this section may become enabled and editable.
        - **Type**: The file format of the Trust Keystore (for example, jks)
        - **Name**: The name of the Trust Keystore file
        - **Password**: The password of the Trust Keystore file (default password for java key store is **changeit**).
- Identity Keystore (For Two Way SSL Only)
  If you are using the Two Way SSL protocol, you must specify the following parameters, which are used only in Two Way SSL:
    - **Type**: The file format of the Identity Keystore (for example, jks)
    - **Name**: The name of the Identity Keystore file
    - **Password**: The password of the Identity Keystore file (the default password for java key store is **changeit**).
- **Additional Parameters**: Use this field to pass any additional SSL parameters.

### Weblogic Cluster Specific Test Server Properties

If you deploy integration modules into a Weblogic cluster, you'll need to enter a Weblogic node address and port against which you want to tun your tests. In a single server configuration, the integration console should point to the single WebLogic server, which is also your Admin server. In a cluster configuration, the integration console should point to one of the node servers in the cluster.



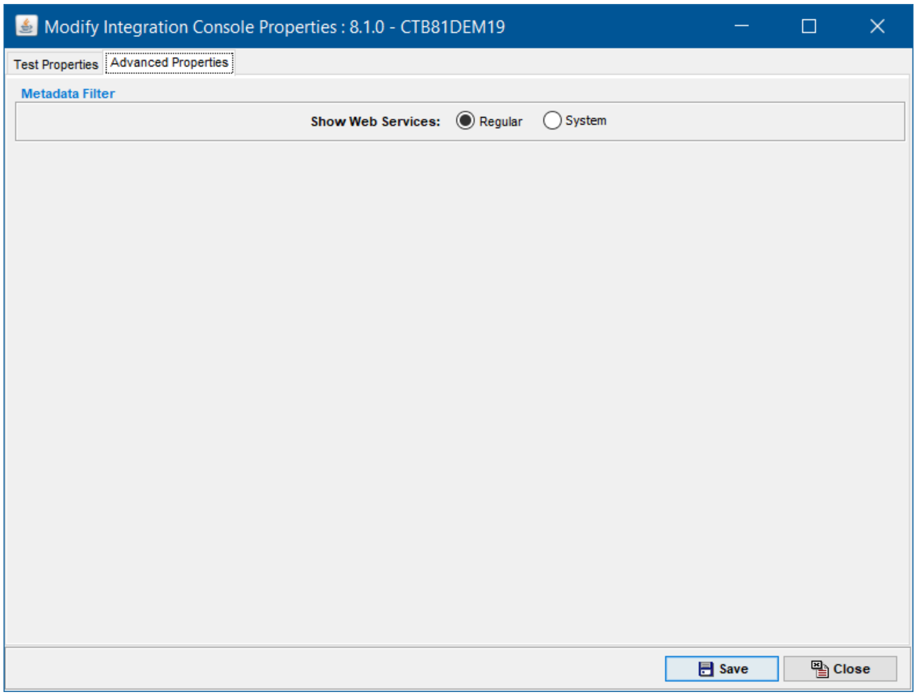
System: C62RADO
User: CPSUPERUSER
Password: ***********
Company: 1
☐ Use Default System
Test Server Properties
IP: alextumanov2xp Port: 7009 SSL Port: 7002

These fields appear at the bottom of the Test Properties tab of the Modify Integration Properties screen:

- **IP**: The IP address of the server in the cluster to be used to test the integration module
- **Port**: The port of the server in the cluster to be used to test the integration module
- **SSL Port**: The SSL Port of the server in the cluster to be used to test the integration module if it has been

configured to use SSL

### Advanced Properties Tab



Show Web Services: ● Regular ○ System

Costpoint contains a few prebuilt Web services that are included in the standard installation or used to do integration with other products (like SFT or MES). These Web services provide interfaces to Deltek partnering products and are considered part of the system. The definition of those modules can be viewed in the Integration Console, but they are not shown by default. Use this screen to control that settings.

- Show Web Services: Select from the options:
- Regular: This is the default option. Only the client-defined Web services are displayed in the Integration Tool.
- System: Use this option to display only the prebuilt Web services built by Deltek. You should not modify these prebuilt Web services as it could break the prebuilt functionality.

# Appendix A: Generic Web Services

Costpoint supports string-based Web services. String-based Web services pass data as XML documents encoded as strings. This means that for string-based Web services, WSDL does not fully describe the structure of input/output documents; they are just strings. To generate a string Web service client, program developers should look at separate XML schema documents that describe the structure of input/output documents. At the same time, it allows you to keep WSDL simple and you can make some changes to the input/output documents structure without changing WSDL.

**Note:** Please note that Deltek de-supported Object-based Web services in versions 8.1 and above, so if you've used Object-based Web Services in prior versions, you will need to convert to Generic Web Services.

For a string-based Web service, Costpoint does not require you to rebuild/redeploy the Web service as long as you only make minor changes to the method structure.

Along with the Web service jar, Costpoint also maintains Web service metadata that it stores in several ADMIN tables and uses in design/run time.

In terms of J2EE application deployment components for string-based Web services, Deltek created two Web Services jars to handle all requests to the generic Web services:

- Cpwwsgenericmodule
- Cpwwsgenericmodulesec

These handle all generic Web service requests over HTTP and HTTPS (One Way SSL). Each module has several methods that handle particular types of Web service requests :

- cpwwsgenericexport: Supports data export Web services
- cpwwsgenericimport: Supports data import Web services
- cpwwsgenericprocess: Supports process Web services
- cpwwsgenericreportnoreturn: Supports report Web services that do not return the report
- cpwwsgenericreporthtmlreturn: Supports report Web services that return report in HTML format
- cpwwsgenericreportpdfreturn: Supports report Web services that return the report in PDF format
- cpwwsgenericreportexcelreturn: Supports report Web services that return the report in Excel format

You define Web service methods the same way as before, but when you create a Web service module, you have an option to create a **Generic Web Service** with string-based parameters passed as clear text. After you define the **Generic Web Service**, you can test it right away without the need to build or deploy it. This is possible because generic web services do not generate 'real' Web services. Costpoint only stores metadata in the system tables that describe the Web service, but to invoke such a Web service, you need to invoke methods of a predefined system Web service (Cpwwsgenericmodule or Cpwwsgenericmodulesec ). At run time, Costpoint looks at the incoming document to figure out which generic web service it belongs to. After that, Costpoint proceeds as if you invoked a normal string-based Web service. When you are ready to move the Web service to the production environment, you need to export it from the test system using the Export functionality of the Integration Console and then deploy it to the production Weblogic server using the Deploy Hotfix utility (or use the Import functionality of the Integration console). If you export only generic web services, the Deploy Hotfix utility will not ask you to restart Weblogic (if it is currently running). You will need to use the Rebuild Global Settings application instead and run the Reload global settings action to refresh the Web service metadata after deploying the Web services.

**Note:** While the cpwwsgenericmodule**sec**WS endpoint is still supported, it is provided now mainly for backward compatibility purposes. You advised to use **cpwwsgenericmoduleWS** endpoint over **tls/ssl/https** connection.

There is no need to use cpwwsgenericmodulesecWS.