IBM ILOG CPLEX Optimization Studio SPSS Connector in CPLEX Studio

Version 12 Release 6

Copyright notice

Describes general use restrictions and trademarks related to this document and the software described in this document.

© Copyright IBM Corp. 1987, 2013

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Trademarks

IBM, the IBM logo, ibm.com, WebSphere, and ILOG are trademarks or registered trademarks of International Business Machines Corp., in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at Copyright and trademark information.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, or service names may be trademarks or service marks of others.

© Copyright IBM Corporation 1987, 2013.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

© Copyright IBM Corp. 1987, 2013

SPSS Connector in CPLEX Studio

The SPSS Connector connects IBM SPSS Modeler to IBM ILOG CPLEX Studio and enables you to read data from an SPSS stream into an OPL tuple set. The SPSS Connector is available on Windows 32-bit and 64-bit platforms.

© Copyright IBM Corp. 1987, 2013

Chapter 1. Prerequisites for using this tutorial

It is assumed that SPSS Modeler is installed and you know how to use it.

- IBM SPSS Modeler must be installed on the same machine as the CPLEX Studio IDE. You can download it from: http://www-01.ibm.com/software/analytics/spss/downloads/.
- You must have a license for SPSS Modeler. A trial license is included with the installation and lasts for 90 days. The license can be renewed at: https://spss.subscribenet.com/control/ibmp/RegisterToAccount.
- This tutorial assumes you have a working knowledge of SPSS Modeler.

  A video of the SPSS Modeler overview can be obtained from: http://www-01.ibm.com/software/analytics/spss/downloads/videos.html.

  A demo and a tutorial on SPSS Modeler can be obtained from: http://www-01.ibm.com/software/analytics/spss/downloads/demos.html.

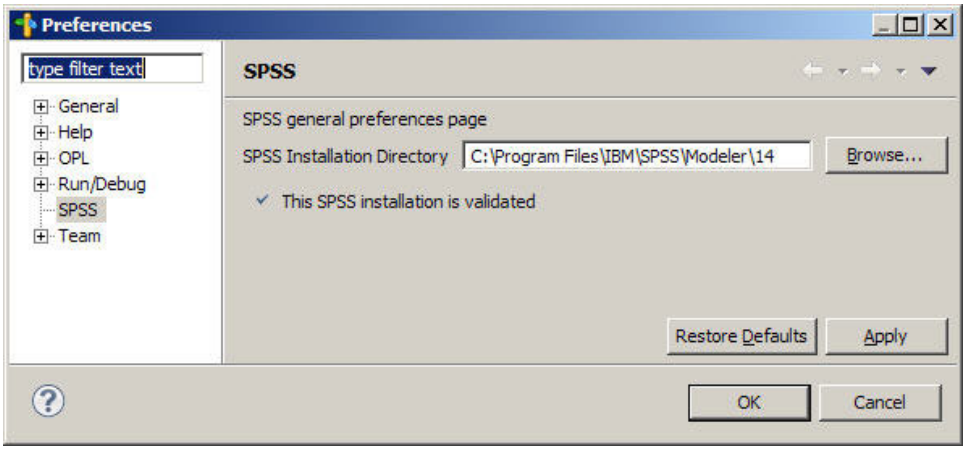
© Copyright IBM Corp. 1987, 2013

Chapter 2. Preparatory steps

Before you can work with an SPSS stream in CPLEX Studio, you need to take some preparatory steps.

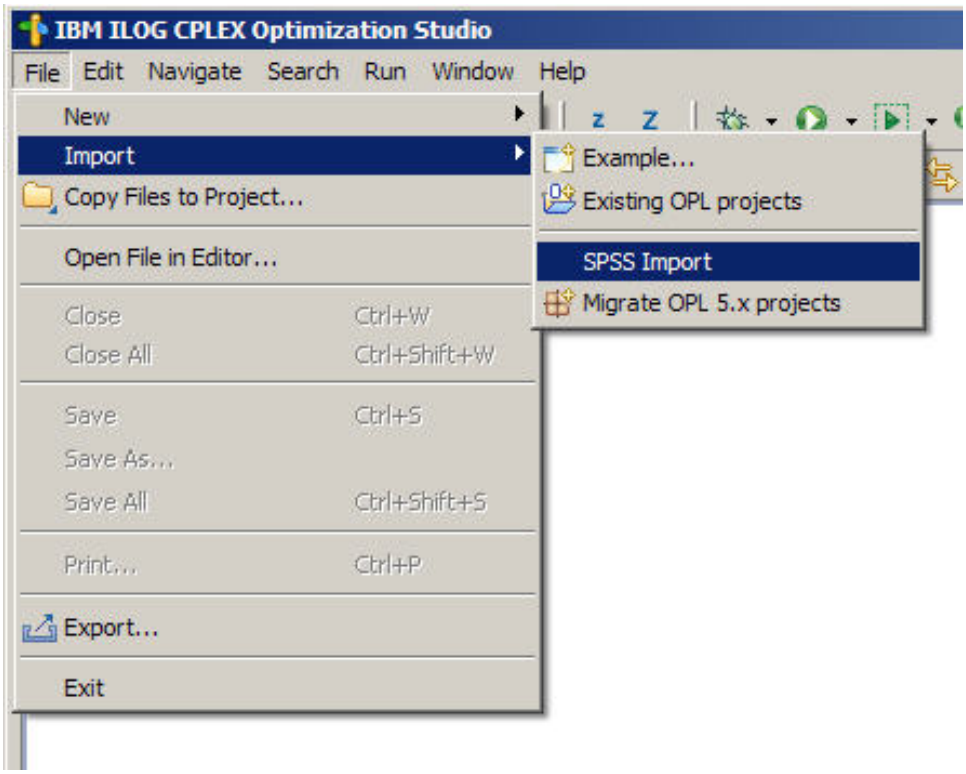
Enabling SPSS Modeler

1. Launch the CPLEX Studio IDE.
2. Ensure that the IDE preferences indicate the installation path for SPSS Modeler. Click **Window** > **Preferences** and select **SPSS**.



3. You should see SPSS in the **Import** list.

© Copyright IBM Corp. 1987, 2013



Setting the SPSS Modeler environment variable

The OPL run time software needs to find a valid SPSS installation. Set the environment variable to refer to your SPSS Modeler installation. For example, on Windows:

Start > **Control Panel** > **System** > **Advanced** > **Environment Variables** > **New**

Variable name: `CLEO_DEMOS`

Variable value: `C:\Program Files\IBM\SPSS\Modeler\14\Demos`

If SPSS Modeler is not installed in the default folder (C:\Program Files\IBM\SPSS\Modeler\14), you need to define the environment variable `OPL_SPSS_HOME` with the alternative installation path as its value.

If this environment variable is absent, OPL will search in the default SPSS installation location.

You are advised to restart the IDE after setting the environment variable, to avoid problems when running the optimization model later in this tutorial.

Importing an SPSS data stream into the IDE

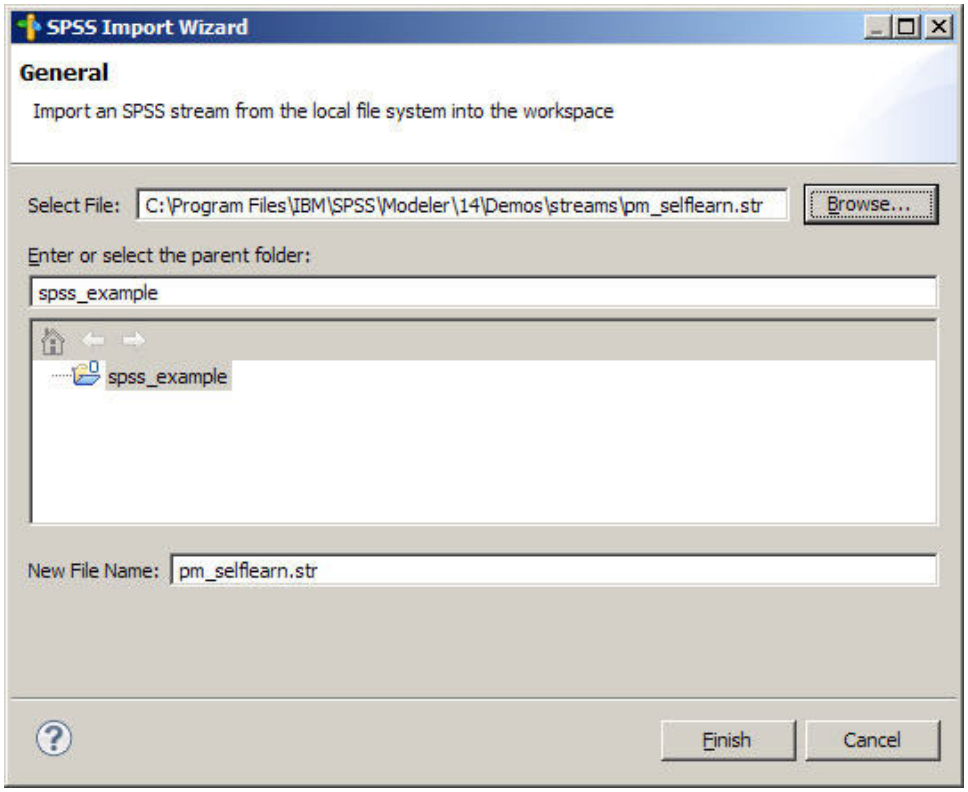
The example stream we use in this tutorial is `pm_selflearn.str`, and can be found in the `Demos` directory of your SPSS Modeler installation. For example:

```
C:\Program Files\IBM\SPSS\Modeler\14\Demos\streams\pm_selflearn.str
```

1. Create a new, empty OPL project in the IDE (**File** > **New** > **OPL Project**) and name it `spss_example`.

Check the box **Add a default Run Configuration**. You will need a default run configuration later in this tutorial.

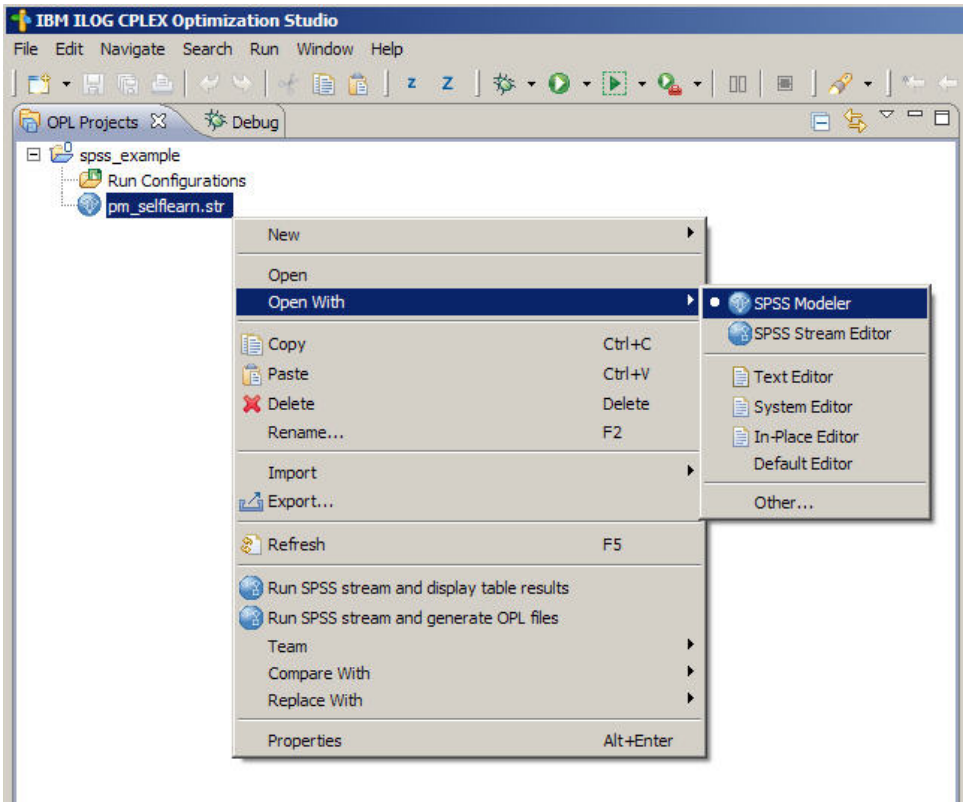
2. Right-click `spss_example` and import the SPSS stream `pm_selflearn.str`.



Adapting the SPSS stream for CPLEX Studio

OPL cannot process SPSS terminal nodes; it processes only operation nodes such as Filter. You must, therefore, modify SPSS streams so that OPL can use them.

1. In the IDE, right-click the stream `pm_selflearn.str` and select **Open With** > **SPSS Modeler**.



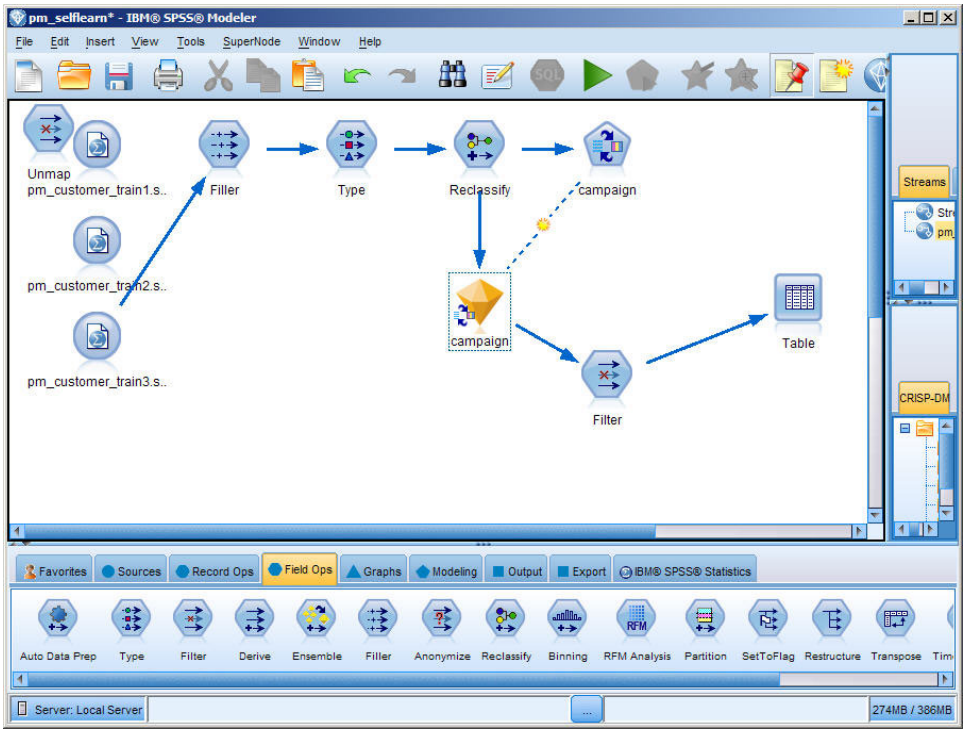
SPSS Modeler opens with the stream `pm_selflearn.str` in the editor.

2. If you run the SPSS stream as it is (run `pm_customer_train3.sav`), it will display a very big result table containing all possible fields. In order to display only the fields of interest for this example, you need to insert an operation node, such as a Filter node, between the campaign node and the result table, and edit the Filter.

   Drag the **Filter** node from the **Field Ops** tab into the editor.

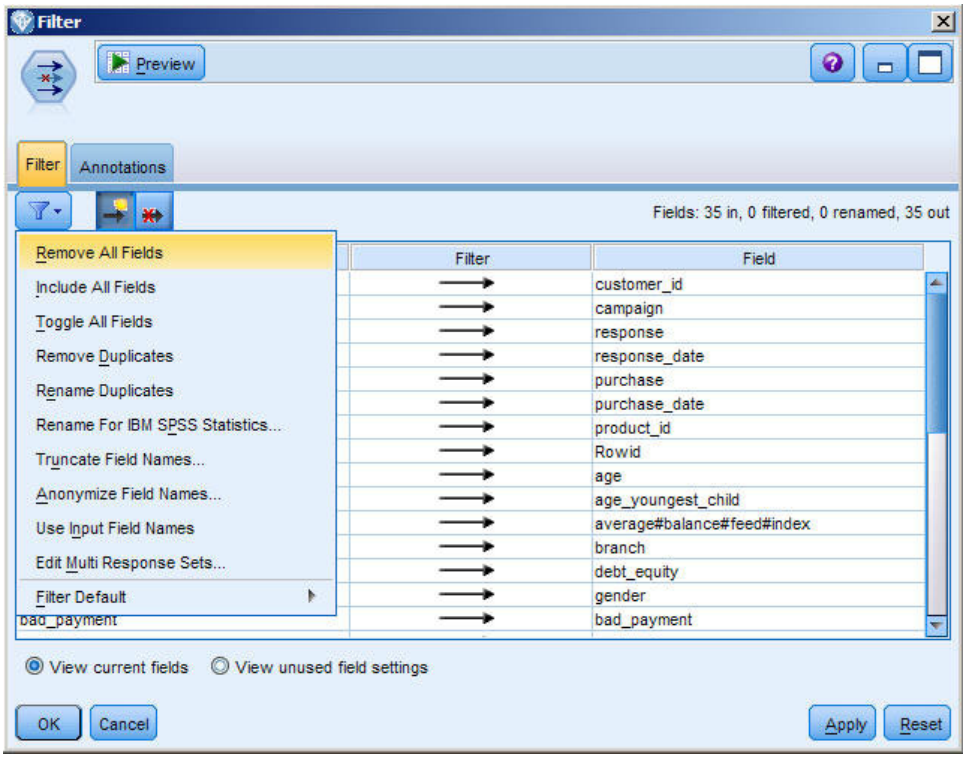
3. Insert the **Filter** node between the **Campaign** node and the terminal node, **Table**. (Select **Campaign** and connect to **Filter**. Select **Filter** and connect to **Table**.)

   The stream now looks like this in the editor.



4. To edit the **Filter** node, right-click the node and select **Edit**, or double-click the node.

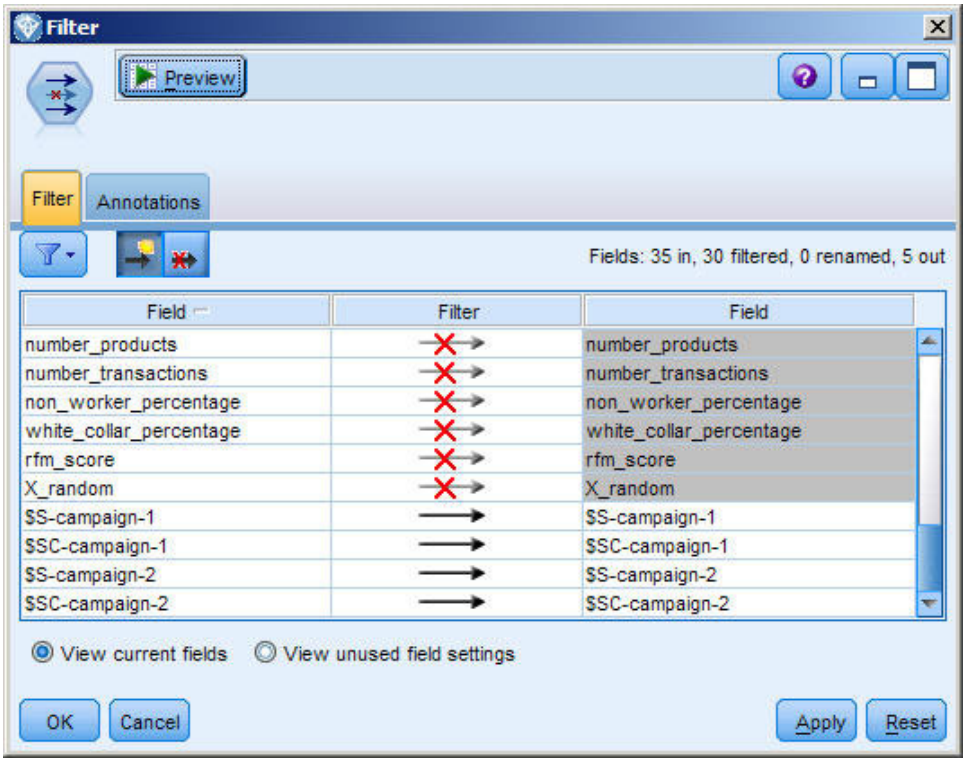
5. Click the **Filter options** icon  to display the list of options and select **Remove All Fields**.



Then click a red cross to select each field you want. For this example, you need the five fields:

`customer_id, $S-campaign-1, $SC-campaign-1, $S-campaign-2, $SC-campaign-2.`

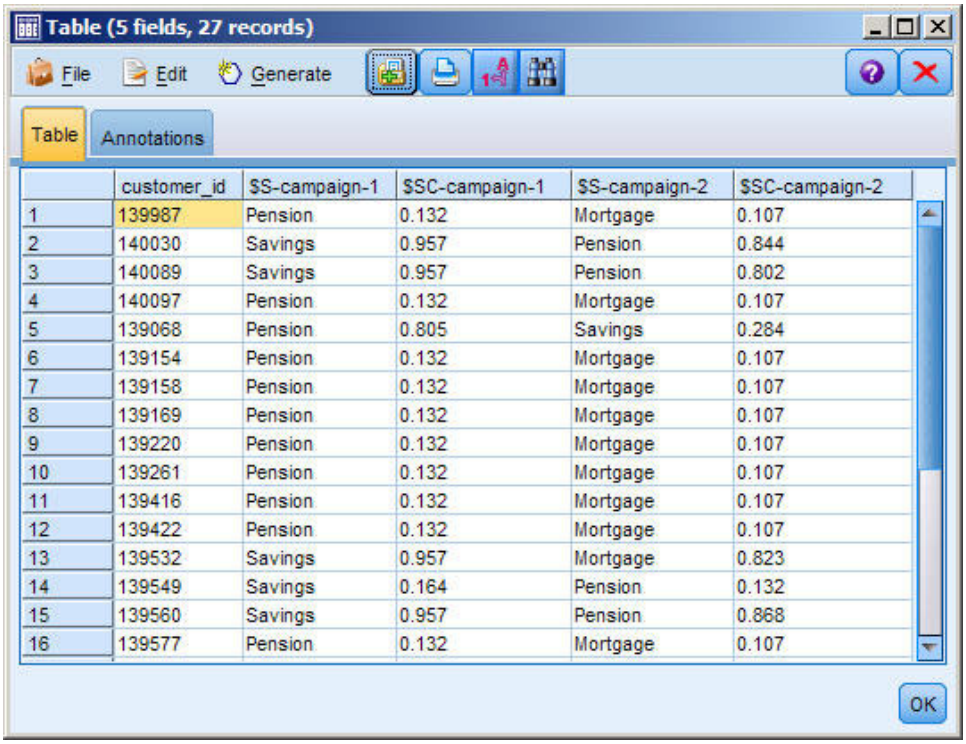
6. Here we see the last four fields, at the bottom of the table, selected.



Click **Apply**, then **OK** to close the Filter node.

7. Run the stream in SPSS Modeler to check that the results table contains only the five required fields.

   Right-click `pm_customer_train3.sav` and select **Run From Here**. The table should look like this:



	customer_id	$S-campaign-1	$SC-campaign-1	$S-campaign-2	$SC-campaign-2
1	139987	Pension	0.132	Mortgage	0.107
2	140030	Savings	0.957	Pension	0.844
3	140089	Savings	0.957	Pension	0.802
4	140097	Pension	0.132	Mortgage	0.107
5	139068	Pension	0.805	Savings	0.284
6	139154	Pension	0.132	Mortgage	0.107
7	139158	Pension	0.132	Mortgage	0.107
8	139169	Pension	0.132	Mortgage	0.107
9	139220	Pension	0.132	Mortgage	0.107
10	139261	Pension	0.132	Mortgage	0.107
11	139416	Pension	0.132	Mortgage	0.107
12	139422	Pension	0.132	Mortgage	0.107
13	139532	Savings	0.957	Mortgage	0.823
14	139549	Savings	0.164	Pension	0.132
15	139560	Savings	0.957	Pension	0.868
16	139577	Pension	0.132	Mortgage	0.107

8. Save the stream and exit SPSS Modeler.

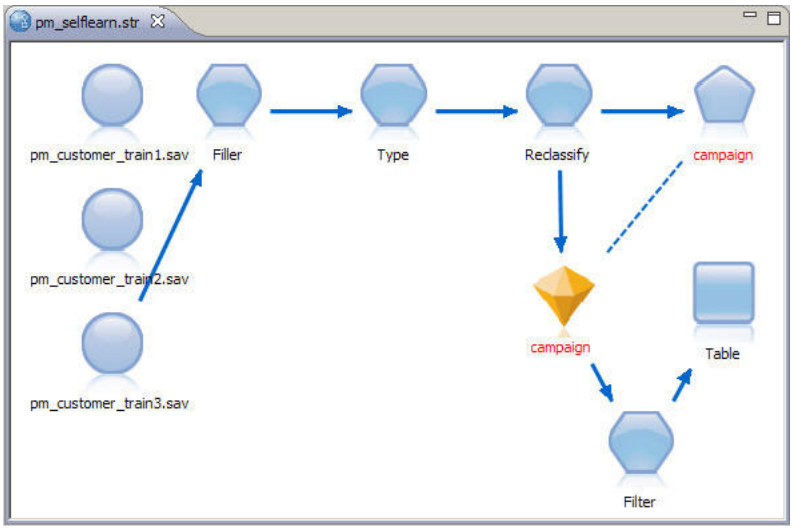
Note:

You cannot modify streams in OPL, so any changes to streams must be made with SPSS Modeler.

Chapter 3. Running an SPSS stream in the IDE

After preparing the environment and adapting the stream, you can run an SPSS data stream in CPLEX Studio.

1. In the IDE, right-click on the stream `pm_selflearn.str` and select **Open With > SPSS Stream Editor** to display the nodes of the stream.



   Note: If several nodes have the same name, the names of those nodes are displayed in red.

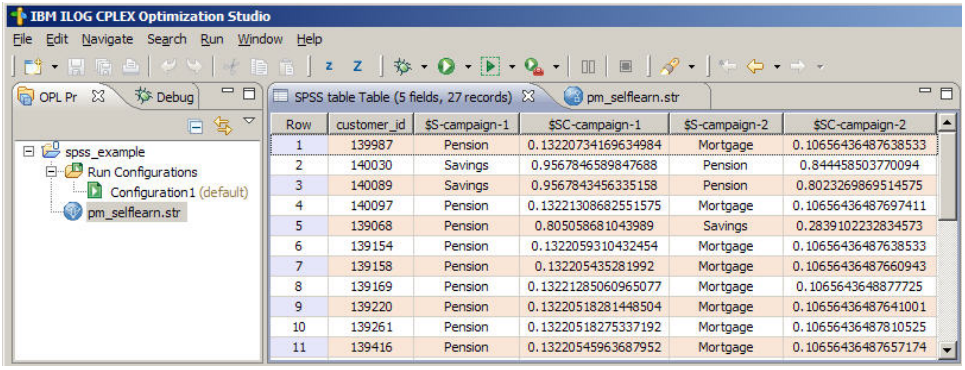
   When the SPSS Stream Editor is open, three zoom buttons become available:


   Zoom Fit, Zoom Out, Zoom In.

2. The following run commands are available in the SPSS Stream Editor when you right-click on a node:
   - **SPSS Run and display results as tables**
   - **SPSS Run and generate OPL files**

   To avoid a warning message in the IDE, you need to have a default configuration in the IDE before executing the command. Earlier in this tutorial we created `Configuration1` as the default configuration.
3. Select the **Table** node and click **SPSS Run and display results as tables for Table** to display a snapshot of the results table in the IDE. This is the results table that was modified in SPSS Modeler.

© Copyright IBM Corp. 1987, 2013

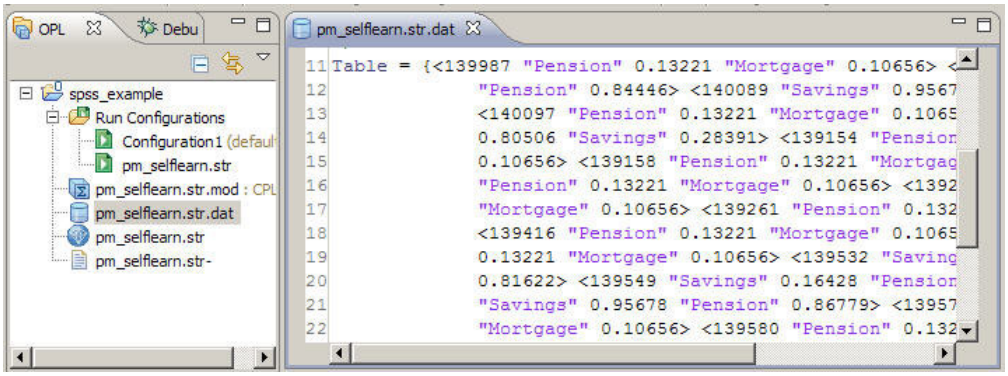


4. Select the **Table** node and click **SPSS Run and generate OPL files for Table**.

   This command copies a snapshot of the results table into an OPL `.dat` file and creates a skeleton of the optimization model. This allows you to work with data from SPSS Modeler in OPL format. A run configuration that uses the two newly-generated files is created automatically.

OPL data file

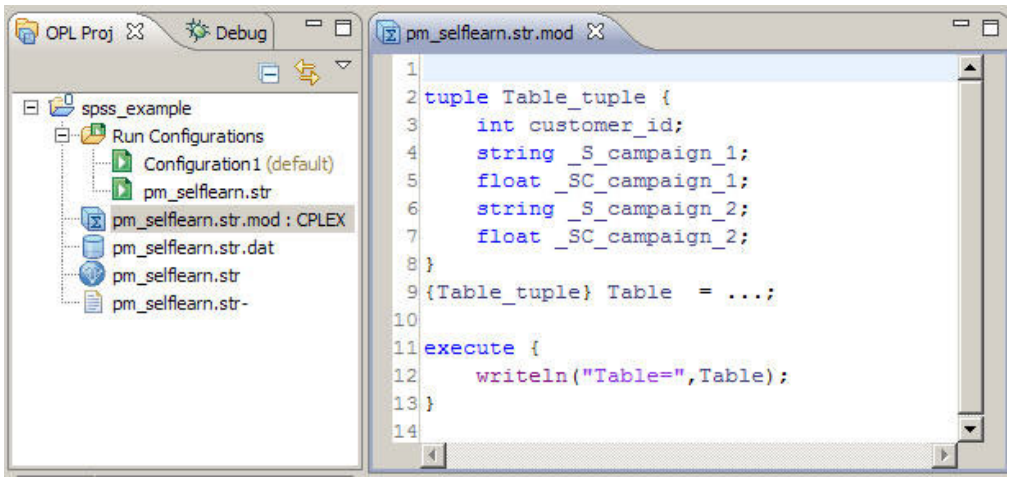
The new data file created, `pm_selflearn.str.dat`, contains the output of the table in OPL format.



OPL model file

The new model file created, `pm_selflearn.str.mod`, contains the definition of the table using tuple schemas.

As OPL cannot process the characters $, &, #, %, -, ., or an empty string, it automatically replaces them with an underscore (_). So, for example, $S-campaign-1 is renamed _S_campaign_1.



With the SPSS Connector, you can create and modify an optimization model in the IDE, and manipulate data from an SPSS stream, without repeatedly launching the stream in SPSS Modeler.

For example, if the SPSS analyst is not the same person as the OR expert, this method provides a way for them to work together. The SPSS analyst continues to work on the stream in SPSS Modeler, while the optimization specialist builds a run configuration in the IDE based on a small fixed data set.

Next step

When the model is ready to go to production, the optimization specialist needs to create a new run configuration with the `.mod` file, and a new `.dat` file containing a connection to the SPSS stream.

The next section explains how to establish the connection with the stream.

Chapter 4. Running an optimization model in the IDE and reading data from SPSS Modeler

Activate the SPSS Connector to run an optimization model in the IDE and read data from an SPSS stream.

1. Create a new run configuration in the IDE, with the skeleton model `pm_selflearn.str.mod` and an empty `.dat` file. In this example, we use the default run configuration `Configuration1` created earlier.

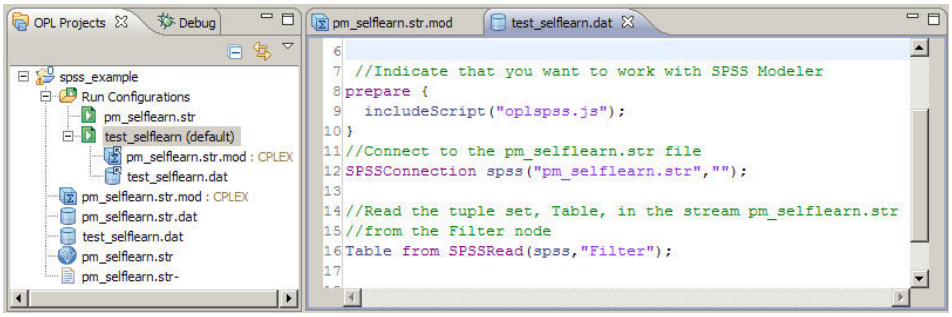
   We rename the configuration `test_selflearn` for clarity.
2. Create a new data file, `test_selflearn.dat`, in which you will enter OPL code to declare that you want to:
   - Work with SPSS Modeler
   - Create a connection between OPL and an SPSS stream
   - Read data from the stream into an OPL tuple set
3. To activate the connection with SPSS Modeler, enter the following OPL code at the beginning of the `.dat` file:

   ```
   prepare {includeScript("oplspss.js");}
   ```
4. Then add the SPSSConnection keyword in the `.dat` file to connect to the stream `pm_selflearn.str` in SPSS Modeler.

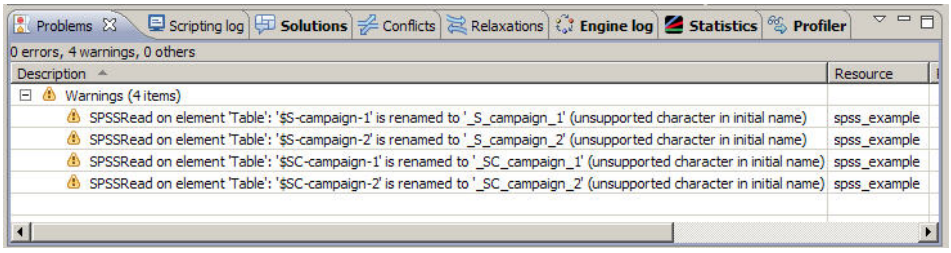
   ```
   SPSSConnection spss("pm_selflearn.str","");
   ```

   The second argument to `SPSSConnection`, the empty string `""`, is for future extensions of the SPSS connector.
5. Add the SPSSRead keyword in the `.dat` file to read the tuple set `Table` in the stream, from the `Filter` node.

   ```
   Table from SPSSRead(spss,"Filter");
   ```
6. The OPL data file, `test_selflearn.dat`, now looks like this:

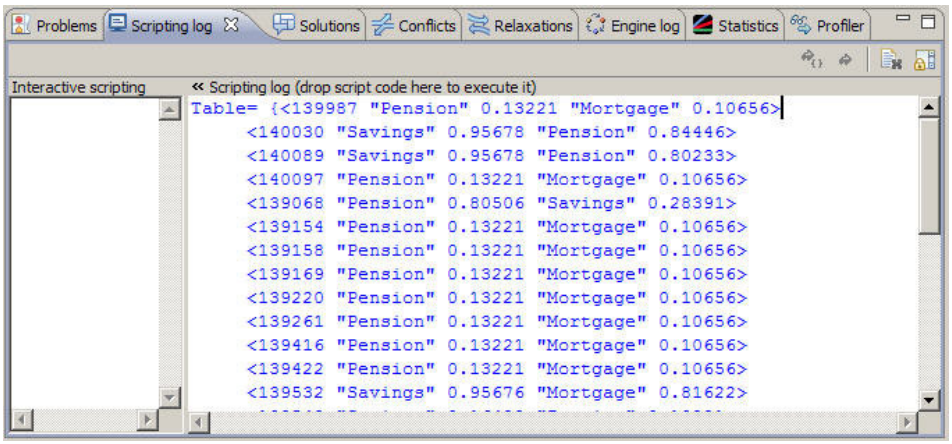


   Save the file `test_selflearn.dat`.
7. Run the configuration `test_selflearn`.
8. The Problems tab displays warnings about the renamed characters.

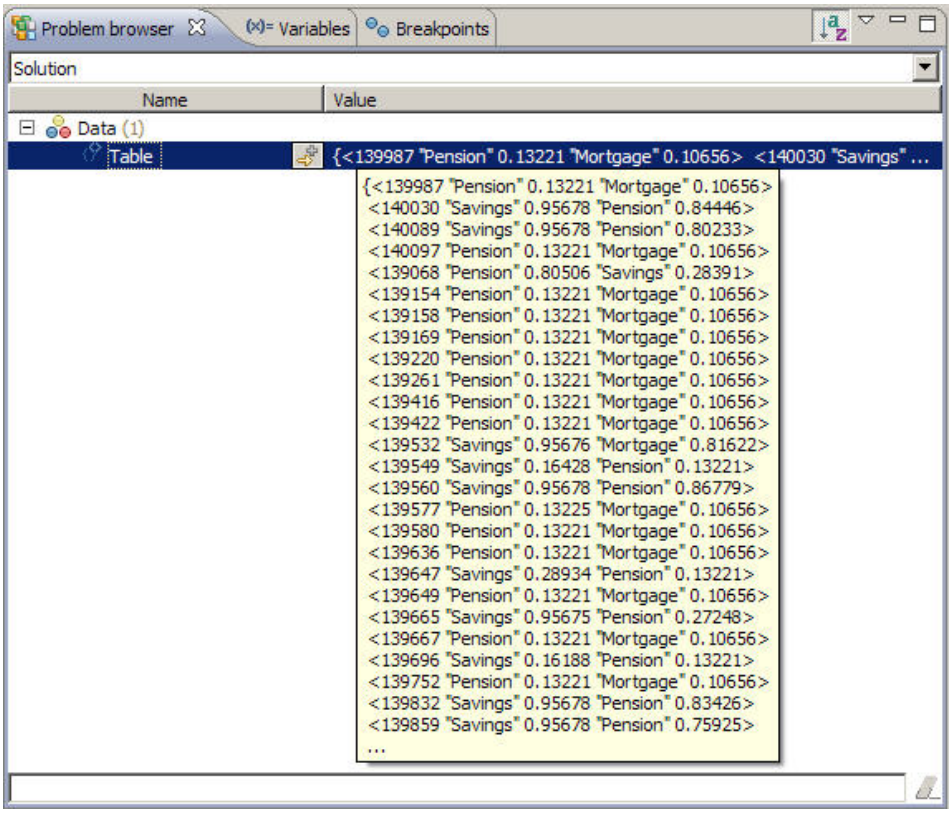


© Copyright IBM Corp. 1987, 2013

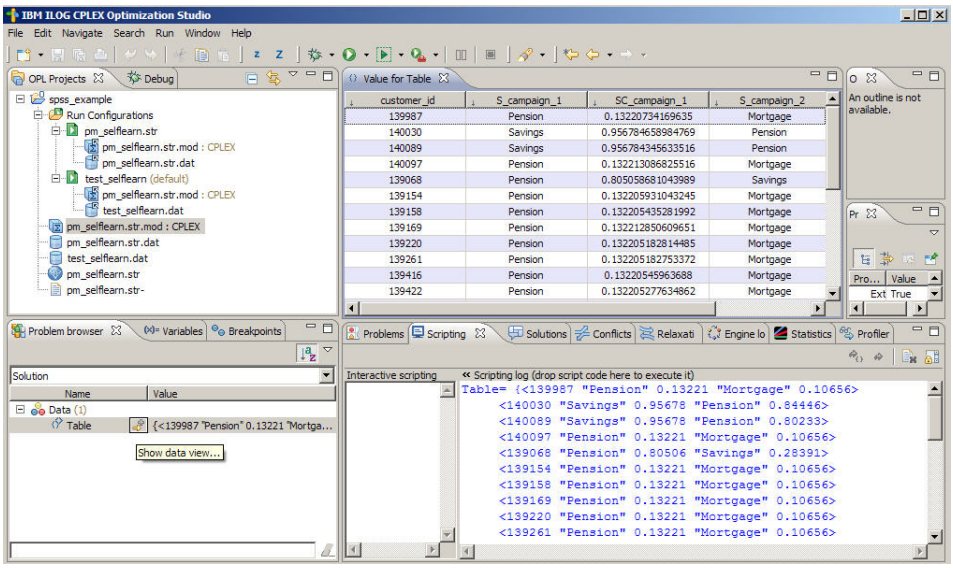
9. The results table is displayed in the Scripting log.



10. In the Problem browser, the results set can be displayed if you hover the mouse over the value cell of the `Table` data item.



11. Finally, you can display the results table in the IDE editor by clicking the **Show data view** button.



From this point you can continue working with your optimization model.

Chapter 5. Getting more information on SPSS stream execution

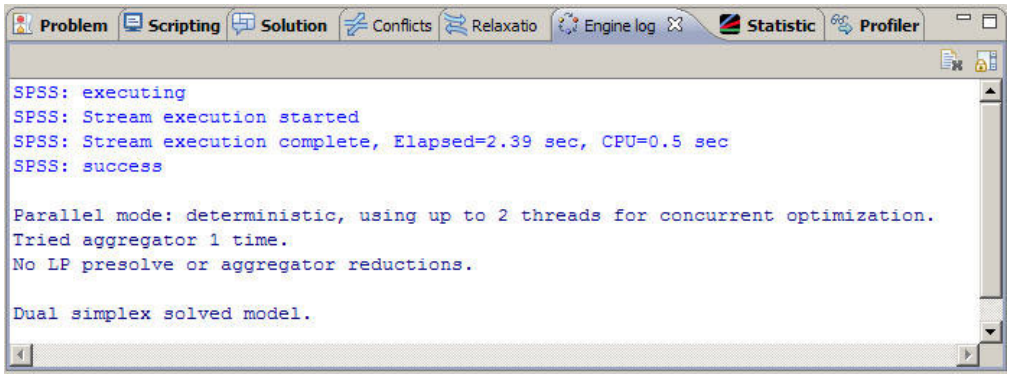
After stream execution, the log from SPSS can be displayed in the engine log of the IDE.

If you execute an SPSS stream in verbose mode, the log information from SPSS Modeler is displayed in the engine log of the CPLEX Studio IDE. To enable the verbose mode, add the statement `SetVerboseSPSS` to the .dat file.

For example, the file `test_selflearn.dat` now contains:

```
prepare {
   includeScript("oplspss.js");
   SetVerboseSPSS();
}
SPSSConnection spss("pm_selflearn.str","");
Table from SPSSRead(spss,"Filter");
```

If you run the configuration `test_selflearn`, the log information from SPSS Modeler is displayed in the engine log of the IDE.



© Copyright IBM Corp. 1987, 2013

Index

© Copyright IBM Corp. 1987, 2013

Printed in USA