Data Express 4.0

Data Subset Extraction

Micro Focus
The Lawn
22-30 Old Bath Road
Newbury, Berkshire RG14 1QN
UK
http://www.microfocus.com

Copyright © Micro Focus 2009-2016. All rights reserved.

MICRO FOCUS, the Micro Focus logo and Data Express 4.0 are trademarks or registered trademarks of Micro Focus IP Development Limited or its subsidiaries or affiliated companies in the United States, United Kingdom and other countries.

All other marks are the property of their respective owners.

2016-04-11

Contents

Data Subset Extraction Guide 5
- Who should read this guide 5
- Data Subset Extraction considerations 5
- Getting Started 6
  - General characteristics of use 6
  - Starting Data Subset Extraction 6
  - Menus 7
  - Context menus 8
- Introduction to Data Subset Extraction window 9
  - Window components 9
  - Node pane 10
  - Column headers by group 11
  - Column headers by creator 12
- Work with Groups 13
  - Introduction to Work with Groups 13
  - Window contents 13
  - Available operations 14
- New Method wizard 14
  - Introduction to New Method wizard 15
  - Creating a new method 15
  - Available operations 17
- Work with Method 17
  - Introduction to the Work with Method area 18
  - Fields 19
  - Steps context menu 19
  - Tabs 19
  - Available operations 22
  - Method confirmation 22
- Work with Method selection class/filter properties 23
  - Introduction to selection class/filter properties 23
  - Window contents 23
- Work with Method work with steps 24
  - Introduction Work with Method work with steps 24
  - Window contents 25
  - Available operations 25
- Work with Method method properties 25
  - Introduction to Work with Method method properties 25
  - Window contents 25
  - Available operations 26
  - Method activation 26
- Work with Method elaboration properties 26
  - Introduction to Work with Method elaboration properties 26
  - Window contents 26
  - Available operations 28
- Options 28
  - Introduction to Options 28
  - Window contents 28
  - Available operations 29
- Data subset extraction jobs 29
  - Introduction to data subset extraction jobs 29
  - Job creation 29

Test environment creation 30
Create allocation jobs 30
JCL information 32
Create unload jobs 32
Create extraction jobs 34
Create reload jobs 37
Available operations 38
Distributed Exporter 39
Window contents 39
Available operations 40
Distributed Statistics Loader 40
Window contents 40
Available operations 41
Work with Method multiple filters 41
Introduction to Work with Method multiple filters 41
More than one single filter for the same data store 41
More than one combined filter involving more than one data element 43

Data Subset Extraction Guide

Outlines the concepts and procedures used by the Data Subset Extraction module. Data Subset Extraction uses selection criteria specified by the user to create a new test environment from an existing environment.

Data Subset Extraction can be used to:

- Generate an integral test environment.
- Generate a reduced test environment.
- Generate a test environment with data migration.
- Generate a test environment while applying data masking techniques (for example, to protect sensitive data).

Data Subset Extraction processing comprises the following elements:

- Methods creation that generates a test environment from the client side.
- Extraction job submission from the client side.

To process an environment using Data Subset Extraction for the Data Express for z/OS solution, all the files belonging to the environment undergo the following steps:

1. Loading and analysis of the application copybooks.
2. Loading and analysis of data stores.
3. Data store/copybook automatic association.
4. Definition and association of selection classes with copybooks.
5. Identification of register table data stores.

For more information, see the *Front End Guide*.

Who should read this guide

This guide is for Micro Focus users who are interested in obtaining a subset of data contained in databases that are either developed internally or purchased by external suppliers.

It explains the Data Subset Extraction configuration procedure, and describes the steps required to migrate the files in your application.

The use of Data Express requires a minimum experience of the mainframe platform z/OS.

Before using this product, we recommend that you carefully review the *Process Guide for z/OS* and *Front End Guide*.

Data Subset Extraction considerations

Bear in mind the following points about the Data Subset Extraction module for Data Express:

- To use Data Express for z/OS, you must have the correct licence. See the *Installation Guide* for more details.
- You do not need to analyze the sources of the programs in the application in order to use Data Subset Extraction.
- The Data Subset Extraction does not perform any operation capable of altering the analyzed data.
- The Data Subset Extraction module manages all types of data stores processed by the Data Builder module with the following exceptions:

- DB2 files with direct access residing in a different DB2 subsystem from the one in which the product is installed.
- DL/I files with direct access

Note: In these two cases, before creating the extraction method, you must specify a sequential unload of the data stores and insert the **Process Identifier** for reading the data stores using the jobs **Load DB2 Table Information by Unload** or **Load DL/I Database Information**. See the *Front End Guide* for more information.

Getting Started

This chapter provides an overview of the Data Subset Extraction module. You should read it before using the module.

General characteristics of use

Data Express for z/OS can be used in either Standard or Client/Server mode, whereas Data Express for Distributed Systems can only be used in Standard mode:

Data Express can be used in either Standard or Client/Server mode:

- Standard - Supports direct access to the data resident on RDBMS from your PC.
- Client/Server - Supports access to the data resident on RDBMS through the Data Express kbde-Server: a product based on a three-tier architecture where the second tier handles data transfer optimization. Client/Server mode lets you access data over the Internet with the lowest amount of administration necessary and the best utilization of network capacity.

  Note: Client/Server mode can use the same functions as Standard mode without you having to install additional software on your PC, such as the DB2 client or the Borland Database Engine (BDE).

For Data Subset Extraction configuration information and instructions, see:

- *Appendix A Standard Mode* in the *Front End Guide*.
- *Appendix B. Client/Server Mode* in the *Front End Guide*.

The following guidelines apply to windows and screen items within the Data Subset Extraction module:

- The main window, **Data Subset Extraction - [Master Form]**, cannot be closed except when exiting the module.
- Multiple MDI windows may be opened simultaneously.
- Right-click an item or area to view a list of applicable commands.

Starting Data Subset Extraction

To start Data Subset Extraction:

1. Click **Start > All Programs > Micro Focus Data Express 4.0 > Data Subset Extraction**.

   Note: If you are using Data Express on Windows Vista and User Access Control is enabled, you must run Data Subset Extraction as Administrator.
2. Select the required database with which to launch the connection. By default, the highlighted item in the list of available databases is the last database you connected to using the Data Subset Extraction module.

   Note: To connect to the database for first time, you need your user ID and password.
3. Select the required schema. The **Data Subset Extraction - [Master Form]** window appears:

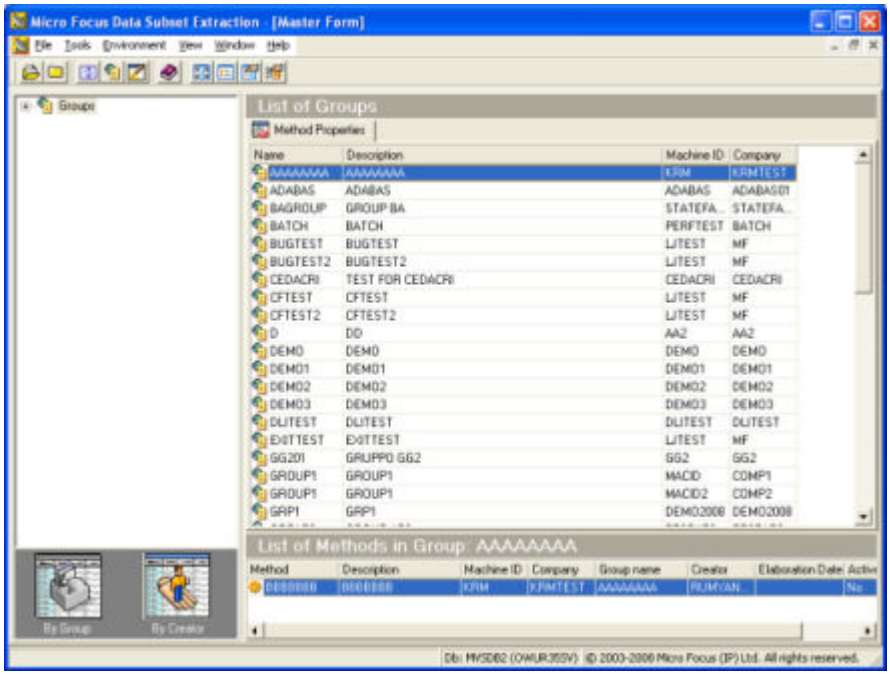

Figure 1: Figure 4-1: Data Subset Extraction - [Master Form] Window

Tip: Once you are connected to a database, you can open and close a database from the **Data Subset Extraction - [Master Form]** window by clicking **File** > **Open** or **File** > **Close**.

Menus

This section describes the menus and common toolbars displayed in the **Data Subset Extraction - [Master Form]** window.

File menu

The **File** menu contains the following commands:

- **Open** - Opens a new database connection.
- **Close** - Closes the database connection.
- **Database - Schema** - Indicates the most recently opened database.
- **Exit** - Terminates the Data Subset Extraction module.

Tools menu

The **Tools** menu contains the following commands:

- **Options** - Opens the **Data Subset Extraction Option** window for setting the help language, security options, and a directory for storing the temporary files generated using the New Method wizard.

Environment menu

The **Environment** menu contains the following commands:

- **Create new Method** - Launches the New Method Wizard so that new methods can be created.
- **Work with selected Method** - Opens the **Work with Method** window containing information for the selected method in order to update or view information.
- **Delete selected Method** - Deletes the selected method.
- **Work with Groups** - Opens the **Work with Groups** window where you can update existing groups and carry out maintenance operations.
- **Work with selected Group** - Opens the **Work with Group** window containing information for the selected group in order to update or view information.
- **Refresh** - Refreshes information.

View menu The **View** menu contains the following commands:

- **Toolbar** - Displays or hides the toolbar.
- **Status bar** - Displays or hides the status bar.

Window menu The set of open windows can be organized according to the standard Windows arrangements. The **Window** menu contains the following commands:

- **Minimize All** - Minimizes all open windows with the exception of the main window.
- **Arrange** - Arranges all minimized windows.

Help menu The **Help** menu contains the following commands:

- **Help Topics** , **Index** - Opens online help information for the product in the language selected in the **Options** window.
- **Micro Focus SupportLine** - Contains commands for connecting to Micro Focus Product Support for information or technical assistance.
- **Micro Focus - Data Home Page** - Connects to the Micro Focus Data Express website.
- **About Data Subset Extraction** - Displays production information for the Data Subset Extraction module.

Context menus

This section describes the commands included in context menus.

Column sorting

A context menu to display a list of sort options can be displayed by right-clicking a column header.

Note: All changes to the grid layout are saved in the Windows system registry. For example, groups , column width attributes, column order attributes , and column font and color information . However, filtering and sorting preferences are not saved.

The column sorting context menu contains the following commands:

- `Sort Descending` - Sorts the list contents in descending order. The same result can be obtained by simply clicking the column heading.
- `Sort Ascending` - Sorts the list contents in ascending order. The same result can be obtained by simply clicking the column heading.
- `Group by This Column` - Groups the list according to the contents of the selected column if the Group Panel has been enabled.
- `Remove This Column` - Deletes the selected column.
- `Column Selector` - Displays a **Customize** window that contains a list of columns that have been removed previously. A column can be restored by performing a drag-and-drop operation to the desired position in the list. A column can be removed from a list by performing a drag-and-drop operation from the list to the **Customize** window.
- `Alignment` - Aligns the content of the columns to the center, right, or left as specified by the user.
- `Best Fit` - Resizes the selected column to allow the content of each record to be displayed. The same result can be obtained by double-clicking the column's external margin.
- `Pick Color` - Allows the color of the selected column to be changed.
- `Pick Font` - Allows the font, font style, size, color, and script to be changed.
- `Best Fit (All Columns)` - Resizes all the columns from the list to allow the content of each record to be displayed.

Note: Columns can also be sorted by clicking or performing a drag-and-drop operation on the column heading. Information displayed can be reduced by clicking a column heading, and then typing the first letter of the items you are interested in.

Grid functions

A context menu to display a list of functions for a grid can be displayed by right-clicking anywhere on the grid.

Note: All changes to the grid layout are saved in the Windows registry. For example, groups , column width attributes, column order attributes , and column font and color information . However, filtering and sorting preferences are not saved.

The grid functions context menu contains the following commands:

- `Column Selector` - Displays a **Customize** window that contains a list of columns that have been removed previously. A column can be restored by performing a drag-and-drop operation to the desired position in the list. A column can be removed from a list by performing a drag-and-drop operation from the list to the **Customize** window.
- `Best Fit (All Columns)` -Resizes all the columns from the list to allow the content of each record to be displayed.
- `Export to File…` - Exports the information to a file using one of the following compatible formats:
  - Comma separated value (`.csv`)
  - Hyper Text Markup Language (`.htm`)
  - Text (`.txt`)
  - Excel (`.xls`)
  - Extensible Markup Language (`.xml`)
- `Show Summary Footer` - Displays the summary footer beneath the grid. By selecting a row and right-clicking the summary footer, a context menu displaying arithmetic operations you can perform on the selected row appears.
- `Show Group Panel` - Displays the column name by which data is grouped above the grid. By dragging the column headers into this window, a group hierarchy can be defined.

  Note: If data is being grouped by column, a footer panel is displayed beneath the grid for each group.
- `Filter Active` - If enabled, lets you choose data to be displayed on the grid; a downward arrow icon will appear in the column headers.

  Click an arrow to choose the data to be displayed on the grid. This function lets you generate a personal filter.

Note: The time taken to display the window increases with the number of records to be loaded.

Introduction to Data Subset Extraction window

After starting the Data Subset Extraction module, the main window **Data Subset Extraction - [Master Form]** appears.

From this window, you can view information regarding existing groups and methods.

Window components

This section describes each area of the **Data Subset Extraction - [Master Form]** window.

The **Data Subset Extraction - [Master Form]** window comprises two main areas: the node pane and the grid area as shown:

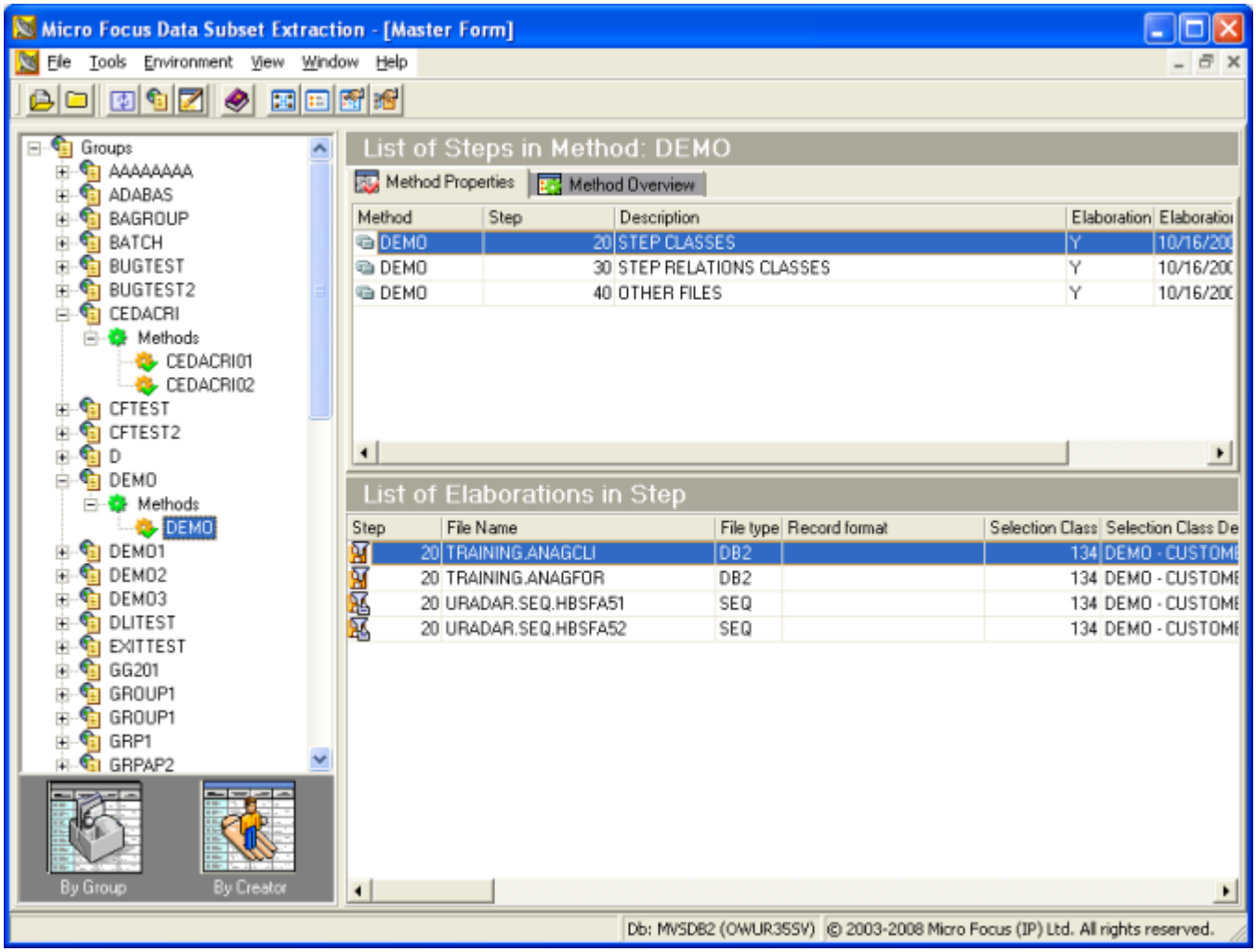

Figure 2: Figure 5-1: Data Subset Extraction - [Master Form] Window

Node pane

The Node pane contains a hierarchical structure that shows the grouping associated with each root-level node.

The two root-level nodes are Groups and Creators.

Icons

Below the hierarchical structure are two icons (**By Group** and **By Creator**) that let you sort information by altering the display of the root node and sub nodes. By default, the hierarchy is sorted by groups.

A. By Group

If the **By Group** icon is specified and the Groups root node is selected, the list of groups included in the environment is displayed in the List of Groups grid, whereas the grid in the lower part of the screen, List of Methods in Group, displays the methods belonging to the group for the selected record. By expanding the Groups root node, the list of the environment groups is visible.

If one of the groups is selected, the List of Methods in Group grid containing environment methods is displayed in the upper part of the screen, whereas the grid in the lower part of the screen, List of Steps in Method, displays the steps making up the method of the selected record.

B. By Creator

If the **By Creator** icon is specified and the Creators root node is selected, the list of creators included in the environment is displayed in the List of Creators grid, whereas the grid in the lower part of the screen, List of Methods created by, displays the methods defined by the creator belonging to the selected record. By expanding the Creators root node, the list of the environment creators is visible.

If one of the creators is selected, the List of Methods created by grid containing environment methods is displayed in the upper part of the screen, whereas the grid in the lower part of the screen, List of Steps in Method, displays the steps making up the method of the selected record.

Context menu

A context menu to display a list of functions can be displayed by right-clicking anywhere in the node pane.

This context menu contains the following commands:

- `Work with selected Method` - Displays the `Work with Method` window including the details for the selected method.
- `Save Method As` - Displays the `Save Method As` window where the method name and description can be changed.
- `Create new Method` - Launches the New Method wizard.
- `Delete selected Method` - Deletes the selected method.
- `Work with selected Group` - Displays the `Work with Group` window for the selected group.
- `Refresh` - Updates information in the node pane corresponding grid areas.

Column headers by group

This section describes the column headings in the Grid area for data stores that have been loaded in the Data Express Knowledge Base when items in the Node pane have been sorted by group.

List of groups

The following list describes the column headings in the List of Groups grid area when the Groups root node is selected:

- `Name` - Name of the group.
- `Description` - Description of the group
- `Machine ID` - Identifier of the machine.
- `Company` - Identifier of the company.

List of methods in group

The following list describes the column headings in the List of Methods in Group grid area for the method selected in the List of Methods in Group grid, when the Groups root node is selected.

- `Method` - Name of the method.
- `Description` - Description of the group.
- `Machine ID` - Identifier of the machine.
- `Company` - Identifier of the company.
- `Group name` - Name of the group.
- `Creator` - Name of user creating the method.
- `Elaboration Date` - Method elaboration date.
- `Active` - Method activated (`Yes` activated, `No` deactivated).
- `Completed` - Method completed (`Yes` completed, `No` not completed).

List of Steps in Method Group

A. Method Properties

The following list describes the column headings in the List of Steps in Method grid area (from the **Method Properties** tab) for the method selected in the List of Methods in Group grid, when either a group node or a method node is selected:

- `Method` - Name of the method.
- `Step` - Number of the method step.
- `Description` - Description of the step.
- `Elaboration` - Elaboration state of the step (`Y` executed, `N` not executed).
- `Elaboration Date` - Step elaboration date.
- `Elaboration Time` - Step elaboration time.

B. Method Overview

The **Method Overview** tab is only visible if the batch job has been already executed for this method.

The bar chart shows the number of the data stores in the selected method and their original and reduced sizes as follows:

- `Reduced Data Stores` - Data stores containing the filter class.
- `Not Reduced Data Stores` - Data stores that do not contain the filter class.
- `All Data Stores` - Sum of the `Reduced Data Stores` and `Not Reduced Data Stores`.

Column headers by creator

This section describes the column headings in the Grid area for data stores that have been loaded in the Data Express Knowledge Base when items in the Node pane have been sorted by creator.

List of creators

The following list describes the column heading in the List of Creators grid area when the Creators root node is selected :

- `Creator` - Name of the creator.

List of methods created by

The following list describes the column headings in the `List of Methods created by` grid area when either a creator node or a method node is selected:

- `Method` - Name of the method.
- `Description` - Description of the group.
- `Machine ID` - Identifier of the machine.
- `Company` - Identifier of the company.
- `Group Name` - Name of the group.
- `Creator` - Name of user creating the method.
- `Elaboration Date` - Method elaboration date.
- `Active` - Method activated (`Yes` activated, `No` deactivated).
- `Completed` - Method completed (`Yes` completed, `No` not completed).

List of steps in method creator

The following list describes the column headings in the List of Steps in Method grid area for the method selected in the List of Methods created by grid, when either a creator node or a method node is selected:

- `Method` - Name of the method.
- `Step` - Number of the method step.
- `Description` - Description of the step.
- `Elaboration` - Elaboration state of the step (`Y` executed, `N` not executed).
- `Elaboration Date` - Step elaboration date.
- `Elaboration Time` - Step elaboration time.

Work with Groups

The **Work with Groups** window lets you to define, view, modify, and delete groups.

Groups can be used to logically separate the methods created for each Machine ID and Company. Each method must belong to a group.

Introduction to Work with Groups

To access the **Work with Groups** window, click (Work with Groups) from the main window.

Window contents

This section describes the columns and fields listed in the Work with Groups window as shown below:

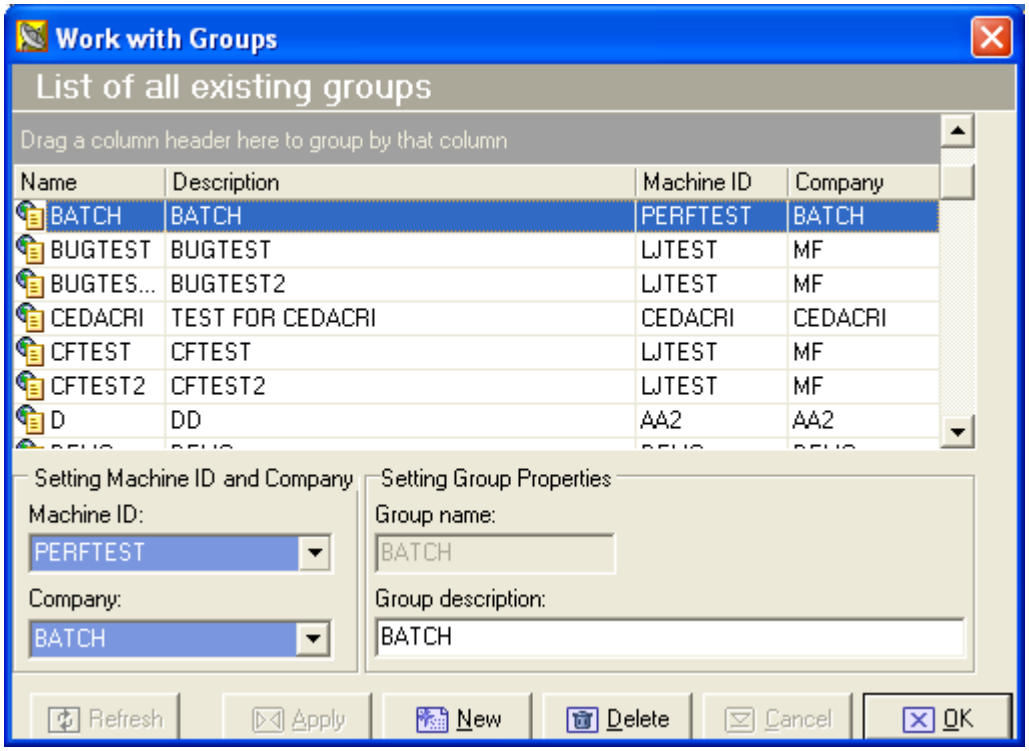

Figure 3: `Figure 6-1: Work with Groups Window`

- `Name` - Identifier of the group. The information in this column matches the information in the`Group Name` field.
- `Description` - Description of the group. The information in this column matches the information in the`Group Description` field.
- · `Machine ID` - Identifier of the machine.
- `Company` - Name of the company.

Available operations

The buttons at the bottom of the **Work with Groups** window are:

- **Refresh** - Refreshes the information listed in the grid. If another user creates a group from a different executable in the same database, it is possible to see the changes.
- **Apply** - Confirms the changes made to the selected group. Only the Group Description field can be modified.
- **New** - Initializes fields for insertion when creating a new group.
- **Delete** - Deletes the selected group if it is not associated with any method.

  Note: All methods contained within the group must first be deleted in the **Work with Methods** window.
- **Cancel** - Cancels any changes made to the selected group. Any actions confirmed with the **Apply** button cannot be restored with **Cancel** but must be entered and confirmed again.
- **OK** - Closes the window.

New Method wizard

The New Method wizard takes user input to generate a new method, which is the extraction proposal containing the set of operations needed for the creation of the test environment. The method is later completed in the Work with Method area.

Introduction to New Method wizard

The Method Creation phase begins with the New Method wizard. During this phase, all the information concerning the method is saved locally in a directory indicated by the user. In the subsequent phase, method confirmation, the method is stored in the Knowledge Base where it is visible and editable by all users.

Reduction can be obtained for data stores of a certain size and/or a particular type. In this case, you must be knowledgeable about the application to determine the appropriate selections. However, the Data Subset Extraction module can recognize and automatically create a first subset containing a series of data stores (called register table data stores) that are initially brought into the new test environment. Identifying register table data stores has the following advantages:

- You can create a method to extract only the register table data store subset, which is normally valid for the creation of any test environment independently of the selection criteria applied. Once this method has been saved and confirmed, it can be run whenever necessary.
- You can create a second data store subset, which filters the register data store subset you created, to produce a data store subset that is reduced even further.

To access the New Method wizard, click (Create New Method) from the main window .

Creating a new method

This section explains how to create a new method using the New Method wizard.

Before creating a method, use Data Builder to complete the following operations:

1. Load and analyze the application copybooks.
2. Load and analyze the data stores.
3. Associate data stores and copybooks automatically.
4. Define and associate the data elements with classes needed for the extraction. For example, define the primary selection class and the related class. Other classes (item class) can be defined and assigned to perform further extraction operations.
5. Make sure that the data stores to be processed are correctly split into the two categories, Register Tables and Others.

For more information about Data Builder, see the *Front End Guide* .

To define a method with the New Method wizard, you must complete 3 steps:

1. Definition of general method characteristics: Inclusion/exclusion of Register Tables data stores.
2. Definition of the primary extraction criteria: Identification of a selection class to divide other data stores into class-sensitive and non-class-sensitive data stores.
3. Definition of relationship criteria: Identification of a connection (foreign key and/or primary key) between related (class-sensitive and non-class-sensitive) data stores.

Note: The **Next** buttons are not enabled if an error occurred while entering data.

Definition of general method characteristics

To define general method characteristics:

1. Click (Create new Method) to display the New Method wizard as shown below:
2. Specify the `Machine ID` and `Company` for the data stores that must be processed.

Note: Data stores belonging to different machine IDs and companies cannot be included.

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

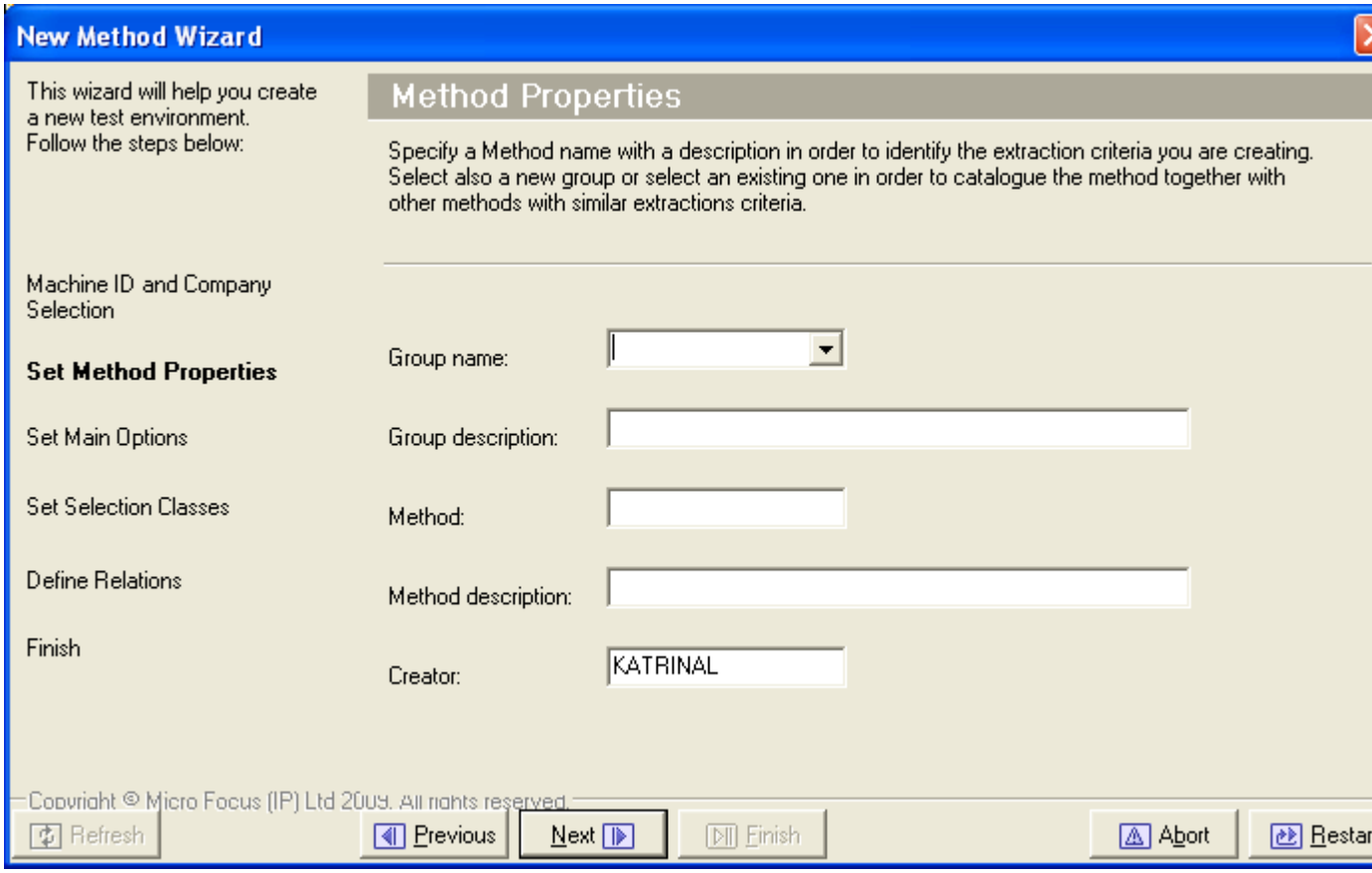

Figure 4: Figure 7-2: New Method Wizard - Set Method Properties

4. Specify the `Group name`, `Method`, `Method description`, and `Creator`.

   Note: If a group has not previously been defined in the Work with Groups area, it can be defined from this screen, by simply specifying its name and description in a specific field.

5. Click **Next** to go to the next screen.
6. Indicate the type of files you want to include in the method by checking one or more of the following:
   - `Import all register table Data Stores` - Demographic data stores are registered in a dedicated elaboration step.
   - `Import all non-register table Data Stores`.
7. If only `Import all register table Data Stores` is checked, click **Finish**. Otherwise, continue to the next section.

Definition of primary extraction criteria

This procedure lets you indicate selection classes, which are classes that you want to apply the filter on in order to obtain a subset of data stores.

To define the primary extraction criteria:

1. Click **Next** to go to the next screen.
2. Select one or more classes that will be used as the primary extraction criteria.

   The Data Subset Extraction module automatically extracts all the environment files containing the selected classes, entering them into a specific step. Once the method has been created, the remaining data stores (that is the data stores that are not sensitive to the selected classes) will initially be excluded

from elaboration. During the Method Confirmation phase, they can be included in the method again. These data stores will then be integrally copied into the new environment generated by the method involved.

3. Continue to the next section.

Definition of relationship criteria

The Data Subset Extraction module automatically identifies all the data stores associated with a related class and includes them in specific subsets. The procedure lets you indicate related classes, which identify the subsets containing all the files associated with one of the identified classes. To subset the data, a filter is applied based on a list/range of values so that all data stores can be automatically copied according to the specified criteria.

Files correlating to those previously filtered may be included in this last subset. These data stores can contain independent information items and, therefore, be integrally copied to maintain the integrity of data. They can also contain strictly correlated information items, such as additional data or document details. Their integral copy will not ensure that data is integrally maintained; that is, the system will include information whose primary origin has not been maintained.

To define the relationship criteria:

1. Click **Next** to go to the next screen.
2. One or more classes related to each class selected as a primary extraction step can be indicated. The primary extraction criteria is listed in the `Selection class` field.
3. Choose the appropriate class in the field **Selection class** and use **Add** to relate additional classes in the Other Class list.

   The Data Subset Extraction module automatically extracts all the environment files containing the related classes and inserts them in a further elaboration step.
4. Click **Finish**. The **Create Method** window is displayed.
5. If required, modify the recommended path where method information will be stored locally, and click **Start**.
6. Click **OK** in the **Generating files list** window to open the Work with Method area.
7. Complete the method by confirming it and by cataloging data stores into elaboration steps in the Work with Method area. For more information, see *Work with Method*.

Available operations

The buttons in the New Method wizard are:

- **Add** - Adds the selected classes in the Other Class list to the Selection Class list.
- **Remove** - Removes the selected class from the Selection class list.
- **Refresh** - Refreshes the information listed for all created classes.
- **Previous** - Goes back to the previous page in the New Method wizard.
- **Next** - Advances one page in the New Method wizard.
- **Finish** - Updates the various tables with information for the new method and saves the elaborations for the new method in a temporary file.
- **Abort** - Exits the New Method wizard without saving changes.
- **Restart** - Restarts the New Method wizard without saving changes.

Work with Method

The Work with Method area lets users confirm newly created methods or modify existing methods at any point after they have been created using the New Method Wizard .

Each newly created method must be completed during the Method Confirmation phase.

Introduction to the Work with Method area

If you have just created a method using the New Method Wizard, the Work with Method area is automatically displayed for the method being defined. Otherwise, to access the Work with Method area, perform the following steps:

1. Launch Data Subset Extraction and access the main window.
2. Locate the method by searching by group or creator in the node pane of the main window. (See the *Node pane* section.)
3. Select the method.
4. Right-click **Work with selected Method**.

   The **Work with Method** window comprises a general fields area, the Steps pane, and the following tabular views:

Method Confirmation phase

During the Method Confirmation phase, specified filters are applied to the primary classes needed to perform the extraction, and the method information is imported from local databases. Each data store subset is cataloged in a specific elaboration step that can occur during any phase.

Before you can confirm the method, you must specify all required connections needed to obtain a further division of data stores into subsets, and you must resolve the problems highlighted in the Warnings tab.

Method elaboration

Method elaboration occurs after the Method Confirmation phase, but can also occur during or after the Method Creation, Method Confirmation, or Method Activation phase. To complete the method, you must decide whether the files that were initially excluded from elaboration during the method creation phase should continue to be ignored by the method, or if the data stores should be included.

After the method has been saved and confirmed, the extraction batch process elaborates steps sequentially. The elaboration of each step can exploit the information obtained during the elaboration of the preceding step. During method elaboration, the following events occur sequentially:

1. Elaboration and integral copy of the data stores recognized as Register Tables.
2. Elaboration of the data stores recognized as sensitive to a selection class, applying the filter specified by the user and copying only the information complying with the required selections. If a related class is included in these data store, all the values of the field associated with the related class are stored in specific lists for all the copied records (in this phase, the related class is also called out class or output class).
3. Elaboration of the data stores recognized as sensitive to a related class, applying the lists of values generated in the preceding step as filter and copying only the information that comply with the data contained in the lists. (In this phase, the related class is also a selection class.)
4. Elaboration and integral copy of the remaining data stores.

Note: The flow described above is confined to a single level of relationships. Normally, there can be connections and interdependence between data stores at several levels.

Method Activation phase

Method activation occurs after the method has been created and confirmed, and most step elaboration has been completed. For instructions, see *Method activation* in the *Work with Method method properties* section.

Extraction Batch Job Launch phase

Methods are created entirely by the client side of Data Subset Extraction. For the Data Express for z/OS solution, the extraction batch job using the created method is performed by the z/OS platform, though the

job is launched from the client side. For more information, see chapter *Work with Jobs* in your *Front End Guide*.

Before launching the extraction job, you must have allocated the output sequential files with names corresponding to those of the output files indicated during the method confirmation phase on the client side.

Fields

The upper part of the window displays fields that contain information for selected method.

The fields listed in the upper part of the Work with Method area are:

- `Machine ID` - Identifier of the machine on which the method is operating.
- `Group` - Name and description of the group to which the method belongs.
- `Company` - Identifier of the company on which the method is operating.
- `Method` - Name and description of the method.
- `Creator` - Name of the user creating the method.
- `Date` - Execution date of the method.
- `Active` - Indicator of whether the method has been enabled.
- `Completed` - Indicator of whether the method has been completed during the confirmation phase.

Steps context menu

This section describes the commands listed by right-clicking in the Steps pane of the Work with Method area.

The steps context menu contains the following commands:

- `Create new Step` - Creates a new step with a proposed step number that is 10 digits greater than the previous step. A number between two existing steps can be used as the step number.
- `Work with Step` - Opens the **Work with Step** window to activate the modification phase for the selected step. Both the step description and the linked documentation file can be modified.
- `Delete Step` - Deletes the selected step. This function is activated only for the steps that do not have assigned elaborations.

Tabs

This section describes each tab of the Work with Methods area. The tabs are:

Elaborations

The Elaborations tab includes the list of elaborations (without incongruence) that process register table files, files containing the primary class, and files containing related classes.

The section describes the column headings and context menu commands accessible from the Elaborations tab.

Column headings tabs

The columns listed on the Elaborations tab of the Work with Method area are:

- `Step` - Number of the step.
- `Data Store Name` - Name of the data store.
- `Record format` - Format of the record.
- `Selected Class Description` - Description of the selected class.

- `Selected Field Name` - Name of the field associated with the input class.
- `In Filter` - Type of filter applied to the selected class (input class).
- `Out Class Description` - Description of the output class.
- `Out Field Name` - Name of the field associated with the output class.
- `Output Filter` - Generated filter code (of the List type).
- `Write File` - Indicator of whether an output file must be created (`Y` create output file, `N` do not create output file) for the current elaboration.
- `Elaborated` - Indicator of whether the file has been elaborated.
- `Number of Written Records` - Number of records written on the output file.
- `Number of Read Records` - Number of records read by the extraction process.
- `Reduction in Bytes` - Number of bytes of the last reduction process (in simulation or in real).
- `Reduction in Percentage` - Percentage of the last reduction process (in simulation or in real).
- `CPU Time (hh:mm:ss)` -CPU time of the last reduction process (in simulation or in real).
- `Elapsed Time (hh:mm:ss)` - Elapsed time of the last reduction process (in simulation or in real).
- `Simulation Date` - Date when the last simulation extraction ran.
- `Real Extraction Date` - Date when the last real extraction ran.

Context menu tabs

This section describes the commands listed by right-clicking a selected file in the Work with Method area from the Elaborations tab. The context menu contains the following commands:

- `Move Selected to` - Moves the selected elaboration to the indicated step. A list of the existing steps is displayed.
- `Add from Excluded Data Stores` - Displays a list of data stores not included in the method that contain the same class specified as the output class of the selected data store. The output filter generated by the selected data store's output class becomes the input filter for those data stores. You can select one or more data stores that will be included in the step following the one of the selected elaboration. If there is no subsequent step, its creation will be automatically requested.
- `Elaboration`
  - `Clear Selection Class` - Deletes the elaboration input class.
  - `Clear Output Class` - Deletes the elaboration output class.
  - `Clear Output and Selection Class` - Modifies the input and output class.
  - `Selection class / Filter properties` - Displays the **List of Class Elaboration** window in order to choose other filter settings for the selected class.
  - `Filter Settings` -Sets, modifies, or displays selection filters on classes.
  - `Copy Filter from` - Applies the filters already defined for the same class in the method to a class included in the file. Unlike the `Connect Existing Filter` command, this associates the class with a copy of the filter associated with a class. Therefore, any modification to this filter will have no impact on the other classes.
  - `Set/Reset Elaborated` - Allows you to change the elaboration status, so you can reset the elaboration flag in order to allow a new elaboration on the same file.

    Note: This is related to the `Run deletion` parameter. If you run an extraction with `Run deletion` active, all files included in the parameter range are elaborated. If `Run deletion` is not active, only files having elaboration flag not set are elaborated.
  - `Delete Elaborated` - Deletes the elaboration of the file.
- `Elaboration Properties` - Displays the **Elaborations Properties** window containing general information on the elaboration of the file and on the method. From this window, you can modify the name of the writing program, the name of the selection program of the records for the file, and the name of the output file.
- `Create New List` - Verifies the existence of one or more classes that can be used as output class in the elaborated file. If the search is successful, it proposes the list of classes.

- `Create New List Using Existing Filter` - Displays the **List of Classes and Filters assigned to the Current File** window. Choose class and filter to use for this new list.
- `Create Output Data Store` - Writes the output data store for the selected elaboration. All elaborations have this setting by default.
- `Don't Create Output Data Store` - Indicator of whether to write the output data store for the selected elaboration.
- `Utility` - Deletes unused filters.

Excluded files

The Excluded Files tab contains the list of files that have been temporarily excluded from expansions, Excluded files do not contain primary or related classes. Files to be involved in new expansions can be taken from this list.

The section describes the column headings and context menu commands accessible from the Excluded Files tab.

Column headings

The columns listed on the Excluded Files tab of the Work with Method area are:

- **Step** - Number of the step.
- **Data Store Name** - Name of the data store.
- **Data Store type** – Type of data store. For example, DB2, VSAM, DL/I, GDG, SEQ, ODBC, or ORA.
- **Record format** - Format of the record.

Context menu

This section describes the command listed by right-clicking a selected data store in the Work with Method area from the Excluded Files tab. The context menu contains the following command:

- `Move Selected To` - Moves the selected expansion to the indicated step. A list of the existing steps is displayed.

Method overview

The Method Overview tab displays general method information and graphics presenting the number of files included in the selected method. This tab is visible only if the extraction batch job has been already completed for this method.

The pie chart shows the distribution of the classes contained in the files of the selected method. By default, it shows the number of files containing classes and those that do not contain any classes.

The bar chart shows the number of the files in the selected method and their original and reduced sizes for the reduced and not reduced files. The Reduced Files are files containing the filter class, Not Reduced Files are files that do not contain the filter class, and All Files is the sum of Reduced Files and Not Reduced Files.

The columns listed on the Method Overview tab of the Work with Method area are:

- **Method Name** - Name and description of the method.
- **Group** - Name and description of the group to which the method belongs.
- **Machine ID** - Identifier of the machine on which the method is operating.
- **Company** - Identifier of the company on which the method is operating.
- **Number of Data stores/tables** - Number of the files or tables included in the method.
- **Size (in GBs)** - Reduced size in gigabytes.

Warnings

The Warnings tab displays the list of elaborations (with incongruence) or ambiguities that must be resolved by the user. For example, an elaboration for a file containing more than one selection class will generate a warning as there can be only one input class for the same file in the same step.

Available operations

The buttons at the bottom of the **Work with Method** window are:

- **Properties** - Displays the **Method Properties** window containing general information about the method. From this window, you can edit the method description, the Method Active state and the path of the Documentation File.
- **Refresh** - Refreshes the window.
- **Copy to Next Step** - Carries the selected expansion forward to the next step.
- **Exclude Elaboration** - Deletes the selected expansion. If the file to be deleted is not included in other steps, it is copied in Excluded Files (if the list is active).
- **Confirm** - Confirms the method. The expansions stored on the local file are transferred to an effective database.
- **OK** - Saves any changes that have not been applied and closes the window.

Method confirmation

This section describes the procedure for confirming a new method. Method confirmation lets you:

- Provide information concerning the names that will be used for the output files of the new environment. During the method confirmation phase, the tool allows you to adopt a single naming mode, which will be set for all the environment files. Once a method has been confirmed, the name of the output file of an expansion can be modified from the **Elaboration Properties** window.
- Define filters on the selection classes.

To confirm the method:

1. In the Work with Method area, click **Confirm**. The **Confirm Method** window is displayed.
2. Assign a naming convention to output files by choosing one of the following options:
   - **Add Prefix** - Adds the prefix specified in the **Prefix** box to the name of the data set for the output file and to the name of the data set for the input file.
   - **Add Suffix** - Adds the suffix specified in the **Suffix** box to the name of the data set for the output file and to the name of the data set for the input file.
   - **String Replacement** - Generates a new data set name for the output file by replacing a part of the data set name the input file with the text specified in the **String to search** box with text in the **Replace with** box.
3. If required, check **Move excluded files into new step** to terminate the creation of the method inserting all the files excluded from the elaborations in the final step of the method.
4. Click **OK** to save your changes or click **Cancel** to discard your changes.
5. If no errors occurred, apply filters to the primary selected class in the **Selection class / Filter properties** window as required (see the chapter *Work with Method - Selection class / Filter properties*). Otherwise, errors have occurred while output file names were automatically assigned, and the names of the files must be manually assigned. The following conditions will generate errors:
   - The file name exceeds 44 characters.
   - Characters to be replaced have not been found in the input name.
   - One or more qualifiers exceed 8 characters.

   When errors occur, the **Reconcile** window is displayed.

6. Edit the file name for each output file listed in the **Reconcile** window. To edit the file name for a single file, select the file, type the new name in the **Unload Output File Name** window, and click **OK**. To edit all file names, click **Modify** to access the file naming definition window and edit the criteria as required.

Work with Method selection class/filter properties

During the Method Confirmation phase, you can apply filters to the primary selected class in the **Selection class / Filter properties** window.

However, the **Selection class / Filter properties** window lets you modify the filters applied to a class even after the method has been confirmed or insert new filters if selection classes are modified or added.

Introduction to selection class/filter properties

To display the **Selection class / Filter properties** window, select the appropriate step in the Work with Method area. Right-click to display the context menu, point to **Elaboration**, and then click **Selection class / Filter properties**.

Window contents

This section describes the columns, fields, and filter types listed in the **Selection class/ Filter properties** window.

Column headings in top grid

The `List of class in elaboration` grid displays the list of primary selection classes needed for method confirmation.

The following list describes the columns listed in the List of Class in Elaboration grid of the **Selection class / Filter properties** window:

- `Class Code` - Class identifier code.
- `Class Name` - Name of the class.
- `Data Element name` - Name of the data element.
- `Data Element Type` - Types of data allowed for the data element.

Global filter fields

The following list describes the fields listed for each filter type in the **Selection class / Filter properties** window:

- `Filter description` - Description of the filter.
- `Documentation path` - File annexed to the filter for documentation purposes.

Filter types

This section describes the filter types that can be applied to the primary classes listed in the List of Class in Elaboration grid in the **Select class/ Filter properties** window.

If you want to apply a filter to a class and there are several primary selection classes, you must select each class from the list and later select the type of filter you want to specify. This operation must be repeated for each class in the list for which you want to apply a filter.

Note: The filter values for date fields must be in the format *YYYYMMDD* or *YYYYMMDD HHHH.MM.SS* where *HHHH* is based on a 24-hour clock.

After associating the correct filter with each class, click **OK**.

Filter type properties

NO FILTER

Selecting the filter type **NO FILTER** does not display additional fields.

FILTER BY RANGE

Displays the following:

From Value	Start value to be searched for.	
To Value	End value to be searched for.	
Range Mode	**Values in range are included**	Considers values within the specified range.
	Values in range are excluded	Considers values outside the specified range.

FILTER BY USER EXIT

Displays the following:

User Exit name	Name of the user program performing the selection.

FILTER BY LIST VALUE

Displays the following:

Values List	One or more values to be selected.	
List Mode	**Values in list are included**	Considers values in the list.
	Values in list are excluded	Considers values outside the list.

FILTER BY FILTERED LIST

Selecting the filter type **FILTER BY FILTERED LIST** does not display additional fields.

FILTER BY EXISTING FILTER

Selecting the **FILTER BY EXISTING FILTER** filter displays a grid where you can perform drag-and-drop operations to group filters by columns.

Work with Method work with steps

From the **Work with Steps** window, you can create new steps for a method and display the characteristics of the existing steps.

Introduction Work with Method work with steps

The **Work with Steps** window can be accessed from different areas of the Data Subset Extraction module, in that it may appear while performing standard operations during method modification.

To display the **Work with Steps** window in display mode, select the appropriate step in the **Steps** pane of the Work with Method area. See the *Introduction to the Work with Method area* section. Right-click **Work with Step**.

To display the **Work with Steps** window in insert mode, select the appropriate step in the Steps pane of the Work with Method area. See the *Introduction to the Work with Method area* section. Right-click **Create new Step**.

Window contents

This section describes the display and insert modes of the Work with Steps window.

Display mode

The **Work with Steps** window in display mode displays information for the selected step. In display mode, only the fields related to the description of the step and to any documentation file associated with it can be modified.

This section describes the **Work with Steps** window in display mode.

- `Step Number` - Number of the step in the method.
- `Step Description` - Description of the step.
- `Link to Documentation File` - File annexed to the step for documentation purposes.
- `Elaboration Date` - Elaboration date of the step.
- `Elaboration Time` - Elaboration time of the step.
- `Elaborated` - Elaboration state of the step (`Y` elaborated, `N` not elaborated).

Insert mode

The **Work with Steps** window in insert mode displays fields that can be defined for the new step you are creating.

This section describes the **Work with Steps** window in insert mode.

- `Step Number` - Number of the step in the method. By default, the tool indicates the progressive number to be associated with the created step. The indicated number is obtained by increasing the number of the last step found in the method by 10.
- `Step Description` - Description of the step.
- `Link to Documentation File` - File annexed to the step for documentation purposes.

Available operations

The buttons at the bottom of the **Work with Steps** window are:

- **OK** - Saves any changes that have not been applied and closes the window.
- **Cancel** - Cancels and changes and closes the window.

Work with Method method properties

The Method Properties window displays all the information concerning the selected method. The method can be activated from this window.

Introduction to Work with Method method properties

To display the **Method Properties** window, click **Properties** from the Work with Method area. For more information, see *Introduction to the Work with Method area*.

Window contents

This section describes the fields listed in the **Method Properties** window.

- `Machine ID` - Identifier of the Machine on which the method operates.

- `Company` - Identifier of the company on which the method operates.
- `Group` - Name and description of the group to which the method belongs.
- `Method` - Name and description of the method.
- `Creator` - Name of user creating the method.
- `Date` - Execution date of the method.
- `Active` - Indicator of whether the method has been activated. Selecting the **Active** checkbox is a prerequisite for the launch of the extraction batch job performed from the z/OS platform. If the check box is dimmed, the method has not been confirmed.
- `Completed` - Indicates whether the method has been completed through the confirmation phase.
- `Documentation File` - File annexed to the method for documentation purposes.

Available operations

The buttons at the bottom of the **Method Properties** window are:

- **OK** - Saves any changes that have not been applied and closes the window.
- **Cancel** - Cancels any changes made to the options. Any actions confirmed with the **Apply** button cannot be restored with **Cancel** but must be entered and confirmed again.

Method activation

To activate the method:

1. From the **Method Properties** window, check **Active**. A **Warning** window is displayed.
2. Click **Yes** and then **OK** to confirm the activation.

Work with Method elaboration properties

The **Elaboration Properties** window displays all the information concerning the selected elaboration and the file involved in the elaboration.

Introduction to Work with Method elaboration properties

To display the **Elaboration Properties** window, select a step in the Files Elaboration list from the Work with Method area. For more information, see *Introduction to the Work with Method area*. Right-click **Elaboration Properties**.

Window contents

This section describes the fields listed in the **Elaboration Properties** window as shown below:

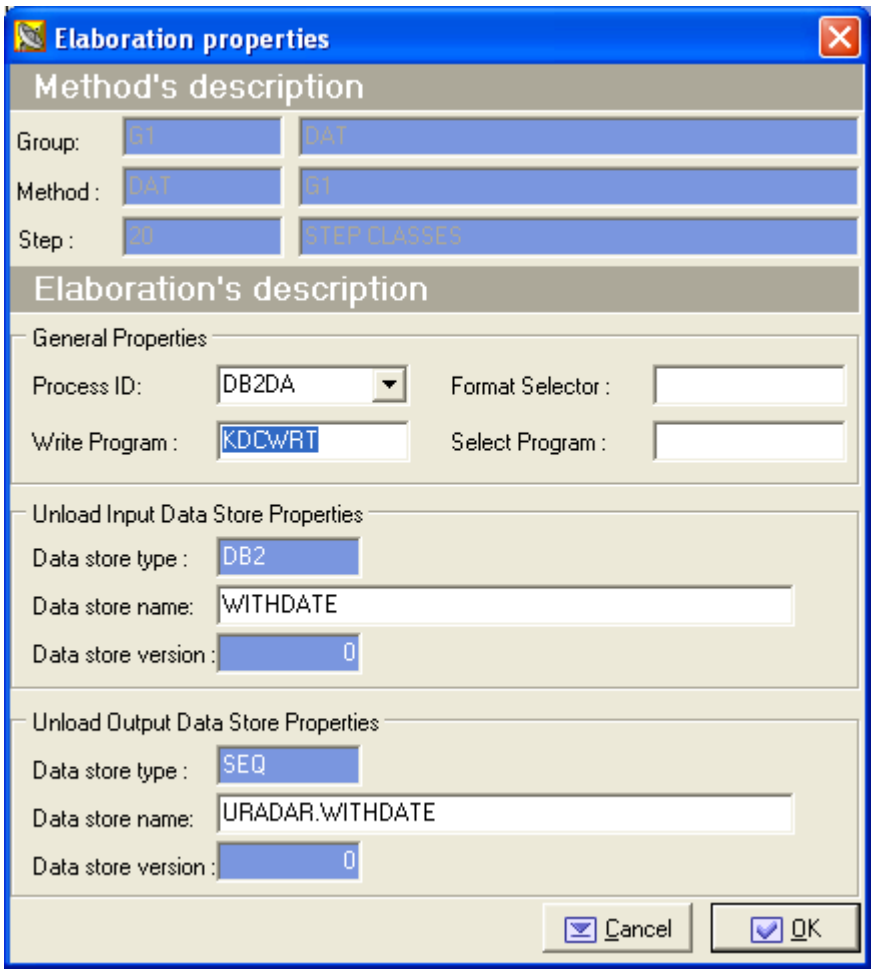

Figure 5: Figure 12-1: Elaboration properties Window

Method’s description

The following list describes the fields listed in the section Method's Description of the **Elaboration Properties** window:

- `Group` - Name and description of the group to which the method belongs.
- `Method` - Name and description of the method.
- `Step` - Number and description of the step of the selected elaboration.

Elaboration’s description

The following list describes the fields listed in the section Elaboration's Description of the **Elaboration Properties** window:

- `General Properties`
  - `Process ID` - Identifier of the process associated with the elaborated file.
  - `Format Selector` - Exit-routine for the recognition of formats associated with the elaborated file.
  - `Write Program` - Write program of the Output file. This information can be modified.
  - `Select Program` - Select program of the individual record for the input file. This information can be modified.
- `Unload Input File`
  - `Data Store name` - Name of the Unload Input data store. This is for the z/OS environment only. It can be different from the original name loaded into the Knowledge Base. In order to do this, the

process ID associated with the elaboration should be created with characteristics for an Unload File Access (For example: GENUN).
    - `Data Store type` - Type of the Unload Input data store.
    - `Data Store Version` - Version of the Unload Input data store.
- `Unload Output File`
    - `Data Store name` - Name of the Unload Output data store.
    - `Data Store type` - Type of the Unload Output data store. This information can be modified.
    - `Data Store Version` - Version of the Unload Output data store.

Available operations

The buttons at the bottom of the **Elaboration Properties** window are:

- **OK** - Saves any changes that have not been applied and closes the window.
- **Cancel** - Cancels any changes made to the options. Any actions confirmed with the **Apply** button cannot be restored with **Cancel** but must be entered and confirmed again.

Options

Data Subset Extraction allows for product customization in order to meet the requirements for different users.

Introduction to Options

Options can be edited in the **Data Subset Extraction** window. To display the **Data Subset Extraction** window, click **Tools** > **Options**. The **Options** window is shown in the section *Window contents*.

Window contents

The **Options** window is divided into pages to help understand the settings.

General

This section describes the fields listed on the **General** tab in the **Data Subset Extraction** window:

- `Help language` - The language used for online help. The Help language list only contains the languages available on your PC.
- `Save user and password` - An indicator of whether or not you want to save your user ID and password information for the current database connection.
- `Select directory` - The directory where unconfirmed methods created with the New Method Wizard are saved.

Server information

Connection options to the kbde-Server for client/server mode can be edited in the **Data Subset Extraction** window. Before opening the project with the Data Subset Extract module, you must define the connection information.

Note: The `Server Information` tab is displayed only in client/server mode.

The following list describes the fields listed on the `Server Information` tab of the `Data Subset Extraction` window:

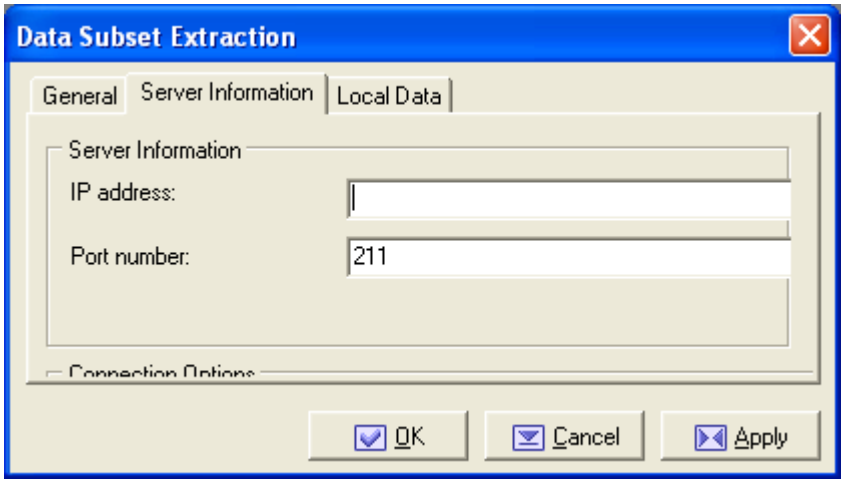

Figure 6: Figure 13-2: Data Subset Extraction - Server Information Tab

- `IP Address` - IP address of the NT server (for `DCom`) or the Borland socket server (for `Socket`).
- `URL` - URL (for `Http`) to identify the `httpsrvr.DLL` file installed on the server machine.
- `Port number` - Port number assigned to Borland socket server (for `Socket`).
- `Connection Options` - the type of connection desired (`Socket`, `DCom`, `Http`).

Available operations

The buttons at the bottom of the **Data Subset Extraction** window are:

- **OK** - Saves any changes that have not been applied and closes the window.
- **Cancel** - Cancels any changes made to the options. Any actions confirmed with the **Apply** button cannot be restored with **Cancel** but must be entered and confirmed again.
- **Apply** - Confirms the change without closing the window.

Data subset extraction jobs

The Work with Jobs area contains information about the function related to the execution of jobs for test environment generation and is part of the Data Builder module. Jobs are submitted using the Work with Jobs area of Data Builder.

Note: Only the Data Express for z/OS solution makes use of Data Subset Extraction jobs.

For more information about the Work with Jobs area, see *Work with Jobs* in the *Front End Guide*.

Introduction to data subset extraction jobs

To access the Work with Jobs area:

1. Start Data Builder by clicking **Start > All Programs > Micro Focus Data Express 4.0 > Data Builder**.
2. Click (Work with Jobs) in the toolbar of the **Data Builder** main window.

Job creation

This section describes the procedures for creating jobs relevant to Data Subset Extraction.

Test environment creation

After all files have been allocated for extraction, launching the Test Environment Creation job is the final step in creating the test environment.

To create a Test Environment Creation job:

1. Create a new Test Environment Creation job from the Work with Jobs area in the Data Builder module. See *Job Creation* in the *Front End Guide* for basic instructions. The **Secondary Options** window for the Test Environment Creation job is displayed.
2. Specify the names of the `Group` and `Method` from the corresponding drop-down lists. If desired, specify the range of steps for which you want to launch the elaboration.
3. Specify the remaining information as appropriate:
   - `Extraction Type`
     - `Access Path Simulation` - Simulates the path to the output files.
     - `Simulation` - If this parameter is set, the generated JCL will run the method in simulation, without populating the output files (doing a first run in simulation is always recommended in order to calculate spaces).
     - `Real Extraction` - If this parameter is set real extraction is performed.
   - `Discard Unchanged Files` - For DB2 tables only, choose this function to discard from the allocation JCL, all DB2 tables belonging to the current method that are neither reduced nor masked.
   - `Run Deletion` - If this parameter is set, a deletion of all results of previous extractions for the same step range is performed.
   - `Run Extraction` - If this parameter is set, the execution of the extractions for the selected step range is performed
4. Click **OK** to submit the job.
5. Wait for the test environment creation phase to be completed.

Create allocation jobs

The Create Allocation job executes a process to create a JCL that will delete and allocate all the output for the data set associated with a method or associated with a business protection (masking) procedure.

Job creation

To create a Create Allocation job:

1. From the **Work with Jobs** window, click **New**.
2. In the List of Jobs section, select **Create Allocation Jobs**.
3. In the General section, select the appropriate `Machine Id` and `Company name`.
4. In the File Qualification section, specify values for the following fields:
   - `Application ID` - The application ID, if applicable.
   - `DB2 Owner, DL/I DBD` - The owner of the DB2 or DL/I DBD file where data will be masked.
   - `File type` - The type of file (`ADA`, `VSAM`, `DB2`, `DL/I`, `GDG`, `SEQ`, or `ORA`).
   - `File name` - The name of the file where data will be masked.
   - `File version` - The version number if the file is GDG.
   - `Record format` - Name of the record format.
5. Click **Apply** to submit the job. The **Secondary Options** window appears:

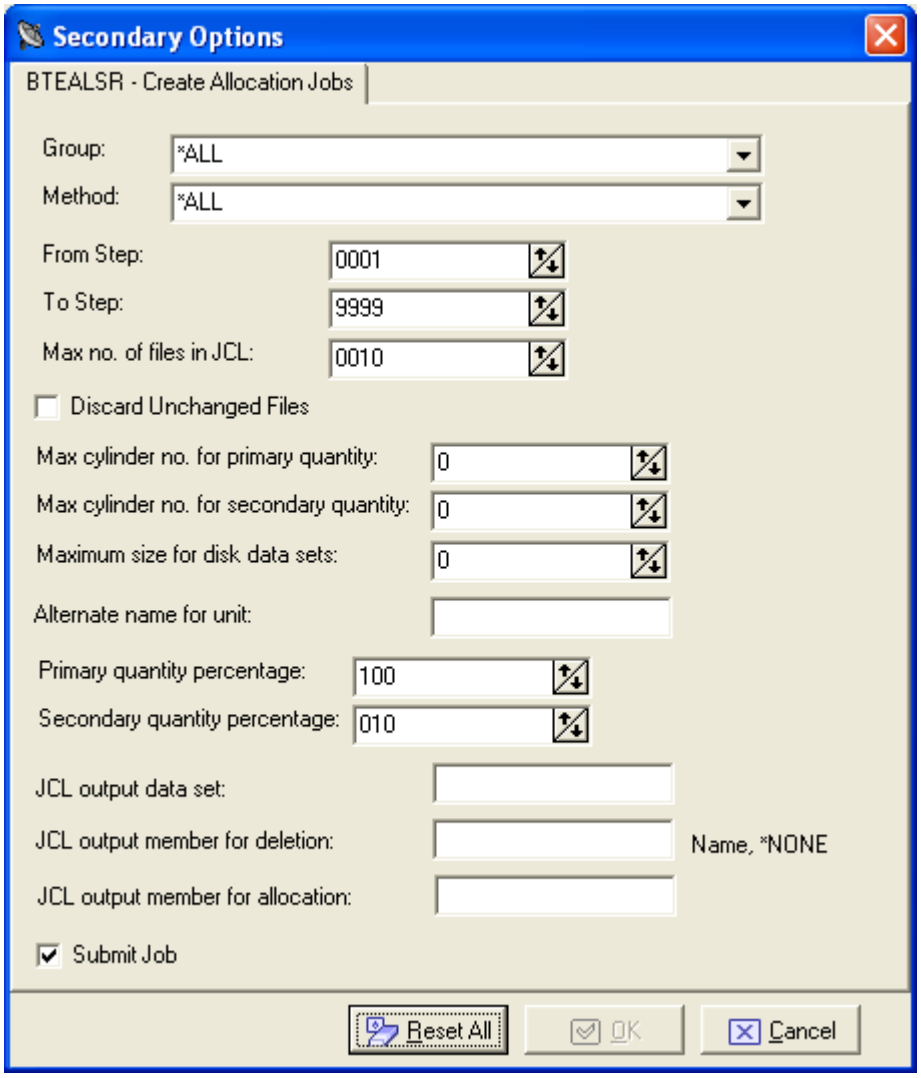

Figure 7: Figure 14-2: BTEALSR - Create Allocation Jobs

6. In the **Secondary Options** window, select the required `Group` and `Method` values, and specify values for the following fields:
   - `From Step / To Step` - Range of values for a subset of steps for the selected method.
   - `Max no. of files in JCL` - Maximum number of files you want process in a single JCL. If your method exceeds this limit, a new JCL is created for the other files. The default value of the parameter is 2000.
   - `Discard Unchanged Files` - Discard all DB2 tables from the allocation JCL that belong to the current method and are neither reduced nor masked.
   - `Max cylinder no. for primary quantity` - Maximum number of cylinders that can be assigned to a primary quantity. The value of 0 indicates to use the default value of 4300 cylinder.
   - `Maximum cylinder No. for secondary quantity` - Maximum number of cylinders that can be assigned to a secondary quantity. The value of 0 indicates to use the default value of 4300 cylinder.
   - `Maximum size for disk data sets` - Maximum number of cylinders (not for the data set) to be allocated on an alternative unit. If the primary quantity exceeds this dimension, the data set will be allocated on an alternative unit.
   - `Alternative name for unit` - Name of the alternative unit where you want to allocate the data set that exceeded the `Maximum size for disk data sets` value.
   - `Primary quantity percentage` - Percentage of cylinder you want to assign at the primary quantity in regard to the total dimension of output data set. The default value is 10%.

- `Secondary quantity percentage` - Percentage of cylinder you want to assign at the secondary quantity in regard to the total dimension of primary quantity. The default value is 15%.
- `JCL output data set` - Library where the generated JCLs will be stored.
- JCL output member for deletion - F irst part of the name of the generated JCLs (the second part will be a 3-digit progressive number starting from 001).
- `JCL output member for allocation` - First part of the name of the generated JCLs (the second part will be a 3-digit progressive number starting from 001).
- `Submit Job` - Indicator of whether to submit the newly created JCLs.

JCL information

The JCLs are built starting from the CBTEALS1 and CBTEALS2 skeletons, which can be customized.

The structure of the skeletons comprises a starting part and an ending part that are repeated once in the created JCLs, and a central part that is delimited by the following keywords:

```
//* SINGLE FILE ELABORATION and
//* SINGLE FILE END ELABORATION, that is repeated for each file.
```

In the skeleton, some variables are used and are then changed into the JCL values using the values stored in the Knowledge Base or included in the launch parameters.

The following table describes the used variables:

Variable	Description
`&TYPE`	Name of the sequential data set (output of extraction process)
`&PCYL`	Primary quantity
`&LRECL`	Record length
`&UNIT`	SYSDA or the value put in the launch screen, depending on the above described rules

`&CYL`	`TRK`
	(if the space is less than a cylinder) or
	`CYL`

`&SCYL`	Secondary quantity
`&JOBN`	It is actualized with the JCL name

Create unload jobs

The Create Unload job executes a process to create a JCL that will unload all DB2 tables (included in the method) with direct access that are not involved in masking and reduction routines.

Job creation

To create a Create Unload job:

1. From the **Work with Jobs** window, click **New**.
2. In the List of Jobs section, select **Create Unload Jobs**.
3. In the General section, select the appropriate `Machine ID` and `Company name`.
4. In the File Qualification section, specify values for the following fields:
   - `Application ID` - The application ID, if applicable.

- `DB2 Owner, DL/I DBD` - The owner of the DB2 or DL/I DBD file where data will be masked.
- `File type` - The type of file (`ADA`, `VSAM`, `DB2`, `DL/I`, `GDG`, `SEQ`, or `ORA`).
- `File name` - The name of the file where data will be masked.
- `File version` - The version number if the file is GDG.
- `Record format` - Name of the record format.

5. Click **Apply** to submit the job. The **Secondary Options** window appears.
6. In the **Secondary Options** window, select the required `Group` and `Method` values, and specify values for the following fields:

- `From Step / To Step` - Range of values for a subset of steps for the selected method.
- `Max no. filesin JCL` - Maximum number of files you want process in a single JCL. If your method exceeds this limit, a new JCL is created for the other files. The default value of the parameter is 2000.
- `Allocate Files` - Indicator of whether to place the generated JCL in the allocation of the sequential output data sets.
- `Max Cylinder no. for primary quantity` -Maximum number of cylinders that can be assigned to a primary quantity. The value of 0 indicates to use the default value of 4300 cylinder.
- `Max Cylinder no. for secondary quantity` - Maximum number of cylinders that can be assigned to a secondary quantity. The value of 0 indicates to use the default value of 4300 cylinder.
- `Maximum size for disk data sets` - Maximum number of cylinders (not for the data set) to be allocated on an alternative unit. If the primary quantity exceeds this dimension, the data set will be allocated on an alternative unit.
- `Alternative name for unit` - Name of the alternative unit where you want to allocate the data set that exceeded the `Maximum Size for Disk Data Sets` value.
- `Primary quantity percentage` - Percentage of cylinder you want to assign at the primary quantity in regard to the total dimension of output data set. The default value is 10%.
- `Secondary quantity percentage` - Percentage of cylinder you want to assign at the secondary quantity in regard to the total dimension of primary quantity. The default value is 15%.
- `JCL output data set` - Library where the generated JCLs will be stored.
- `JCL output member for deletion` - F irst part of the name of the generated JCLs (the second part will be a 3-digit progressive number starting from 001).
- `JCL output member for allocation` - First part of the name of the generated JCLs (the second part will be a 3-digit progressive number starting from 001).
- `Submit Job` - Indicator of whether to submit the newly created JCLs.

JCL information

The JCLs are built starting from CBTUNLD1 skeleton, which can be customized. The CBTUNLD2 skeleton is an alternative skeleton where the UNLOAD utility is used instead of the DSNTIAUL utility.

The structure of the skeleton comprises a starting part and an ending part that are repeated once in the created JCLs, and a central part that is delimited by the following keywords:

```
//* SINGLE FILE ELABORATION and
//* SINGLE FILE END ELABORATION, that is repeated for each file.
```

In the skeleton some variables are used, and they are changed into the JCL values using the values stored in the Knowledge Base or included in the launch parameters.

The following table describes the used variables:

Variable	Description
&NUM	Numeric progressive

Variable	Description
&UNLOUTNAM	Name of the sequential data set (output of extraction process)
&PCYL	Primary quantity
&LRECL	Record length
&UNIT	SYSDA or the value put in the launch screen, depending on the above described rules
&CYL	It can be TRK (if the space is less than a cylinder) or CYL
&SCYL	Secondary quantity
&OWNAME	Owner of the original table
&FILNAME	Name of the original table
&DBNAME	Database of the original table
&TSNAME	Tablespace of the original table
&UNLINPNAM	The unload name as it can be seen in "Work with files"
&STEP	The step name (automatically generated)
&JOBN	It is actualized with the JCL name

Create extraction jobs

The Create Extraction job executes a process to create a JCL that will execute an extraction method.

Job creation

To create a **Create Extraction** job:

1. From the **Work with Jobs** window, click **New**.
2. In the List of Jobs section, select **Create Extraction Jobs**.
3. In the General section, select the appropriate **Machine ID** and **Company name**.
4. In the **File Qualification** section, specify values for the following fields:
   - **Application ID -** The application ID, if applicable.
   - **DB2 Owner, DL/I DBD -** The owner of the DB2 or DL/I DBD file where data will be masked.
   - **File type -** The type of file (`ADA`, `VSAM`, `DB2`, `DL/I`, `GDG`, `SEQ`, or `ORA`).
   - **File name -** The name of the file where data will be masked.
   - **File version -** The version number if the file is GDG.
   - **Record format -** Name of the record format.
5. Click **Apply** to submit the job. The **Secondary Options** window appears.
6. In the **Secondary Options** window, select the required **Group** and **Method** values, and specify values for the following fields:
   - **From Step / To Step -** Range of values for a subset of steps for the selected method.
   - **Extraction Type**:
     - **Access Path Simulation -** Indicator for the generated JCL to run the method in simulation access path.
     - **Simulation -** Indicator for the generated JCL to run the method in simulation, without populating the output files. (A first run in simulation is recommended in order to calculate spaces).

- **Real Extraction -** Indicator to perform real extraction.
- **Discard Unchanged Files -** Discard all DB2 tables from the allocation JCL that belong to the current method and are neither reduced nor masked.
- **Run Deletion -** Indicator of whether to perform a deletion of all results of previous extractions for the same step range.
- **Run Extraction -** Indicator of whether to perform the execution of the extractions for the selected step range.
- **JCL output data set -** Library where the generated JCLs will be stored.
- **JCL output member -** N ame of the generated JCLs.
- **Temporary data set prefix -** Name of the prefix for the temporary data set used in the JCL in order to create the temporary unloads from image copy.
- **Submit Job -** Indicator of whether to submit the newly created JCLs.

JCL information

The JCLs are built starting from *CBTESUB2* skeleton, which can be customized. The structure of the skeleton comprises a single part that is repeated once in the created JCLs, and is delimited by the following keywords:

```
//* UNLOAD ELABORATION and
//* UNLOAD END ELABORATION, by the keywords:
//* SINGLE FILE ELABORATION and
//* SINGLE FILE END ELABORATION and by the keywords:
//* DELETE ELABORATION and
//* DELETE END ELABORATION is repeated for each file.
```

In the skeleton some variables are used, and they are changed into the JCL values using the values stored in the Knowledge Base or included in the launch parameters.

The following table describes the used variables:

Variable	Description
&NAMEJOB	Name of the submitted job
&USER	User who is submitting the job
&URDB2	DB2 subsystem where the product is installed
&URPLAN	DB2 plan of the product
&URLOAD	DB2 load library
&URRUNLIB	DB2 runlib library
&URSRUN	sceerun library
&URPROJ.	First qualifier of the product libraries
&URGRPER.	Second qualifier of the product customization libraries
&URGRPTF.	Second qualifier of the product PTF libraries
&URGRGROUP.	Second qualifier of the product libraries
&NAMEPGM	Name of the first program of the batch chain
&TNSRLNBR	Input parameter `Machine`

Variable	Description
&TNCDSOC	Input parameter `Company`
&TNMETHOD	Input parameter `Method`
&TNGROUP	Input parameter `Group`
&TNSTEP	Input parameter `From Step`
&TNTOSTEP	Input parameter `To Step`
&TNSIMUL	Input parameter `Simulation`
&TNDISFIL	Input parameter `Discard Unchanged Fields`
&TNRUNDEL	Input parameter `Run deletion`
&TNRUNEXT	Input parameter `Run extraction`
&DDNAME	DD name automatically generated by the product
&FILNAME	Name of the unload file generated from image copy
&LABEL	Progressive number
&RECL	Record length for the unload file generated from image copy
&UROWNER	DB2 owner of the product tables
&URPROJ	First qualifier of the product libraries
&URGROUP	Second qualifier of the product libraries
&IMGCPY	Image copy data set
&TYPE	Unload file generated from image copy
&OWNAME	Owner of the original table
&TBNAME	Name of the original table
&DBNAME	Name of the database of the original table
&TSNAME	Name of the tablespace of the original table
&PQTY	Primary quantity
&CYL	*TRK* (if the space is less than a cylinder) or *CYL*
&SCYL	Secondary quantity

Variable	Description
&URDPREFA	First part of the temporary data set names
&URDPREFB	Second part of the temporary data set names
&JOBN	It is actualized with the JCL name

Create reload jobs

This function allows you to create a JCL to reload all DB2 tables with direct access included in a method or associated with a business protection (Masking) procedure.

Job creation

To create a `Create Reload` job:

1. From the **Work with Jobs** window, click **New**.
2. In the List of Jobs section, select **Create Reload Jobs**.
3. In the General section, select the appropriate `Machine ID` and `Company name`.
4. In the File Qualification section, specify values for the following fields:
   - `Application ID` - The application ID, if applicable.
   - `DB2 Owner, DL/I DBD` - The owner of the DB2 or DL/I DBD file where data will be masked.
   - `File type` - The type of file (`ADA`, `VSAM`, `DB2`, `DL/I`, `GDG`, `SEQ`, or `ORA`).
   - `File name` - The name of the file where data will be masked.
   - `File version` - The version number if the file is GDG.
   - `Record format` - Name of the record format.
5. Click **Apply** to submit the job. The `Secondary Options` window appears.
6. In the **Secondary Options** window, select the required `Group` and `Method` values, and specify values for the following fields:
   - `From Step / To Step` - Range of values for a subset of steps for the selected method.
   - `Max no. files in JCL` - Maximum number of files you want process in a single JCL. If your method exceeds this limit, the function creates a new JCL for the other files. The default value is 2000.
   - `Replace Data` - Indicator of whether data replacement in the target database is enabled. If not selected, data items are added without deletion.
   - `Create Syspunch` - Indicator of whether to create the part of the JCL performing the SYSPUNCH .
   - `Create Reload` - Indicator of whether to create the part of the JCL performing the reload.
   - `Target DB2 Owner` - Name of the target DB2 owner. If this parameter is not specified, this value is the same as the source DB.
   - `Target DSN` - Name of the target database subsystem. If this parameter is not specified, this value is the same as the source DB.
   - `JCL output data set` - Library where the generated JCLs will be stored.
   - `JCL output member for reload` - Name of the generated JCLs used for reload.
   - `JCL output member for repair` - Name of the generated JCLs used for repair.
   - `Submit Job` - Indicator of whether to submit the newly created JCLs.

JCL information

The JCLs are built starting from CBTELOD1 and CBTELOD2 skeletons, which can be customized. The CBTUNLD2 skeleton is an alternative skeleton where the UNLOAD utility is used instead of the DSNTIAUL utility in order to create SYSPUNCHs .

The structure of the skeleton comprises a starting and ending part that are repeated once in the created JCLs, and a central part that is delimited by the following keywords:

```
//* SINGLE FILE ELABORATION and
//* SINGLE FILE END ELABORATION, that is repeated for each file.
```

In the skeleton some variables are used, and they are changed into the JCL values using the values stored in the Knowledge Base or included in the launch parameters.

The following table describes the used variables:

Variable	Description
`&NUM`	Numeric progressive
`&UNLOUTNAM`	Name of the sequential data set (output of extraction process)
`&OWNAME`	Owner of the original table
`&FILNAME`	Name of the original table
`&DBNAME`	Database name of the original table
`&TSNAME`	Tablespace name of the original table
`&UNLINPNAM`	Name of the unload as specified in the Work with files area
`&STEP`	Step name for the SYSPUNCH generation (automatically generated)
`&STP1`	Step name for the MISYSPUN program call (automatically generated)
`&STP2`	Name of the reload step (automatically generated)
`&SYSPARM`	It can be ADD,YES, *number of sortkeys* if no replace parameter, REP,YES, *number of sortkeys* if replace parameter
`&DSNNAME`	Name of the input DSN
`&METHOD`	Name of the method
`&JOBN`	Name of the job actualized with the JCL name

Available operations

The buttons at the bottom of the **Secondary Options** window are:

- **Reset All -** Resets all parameters to their default values.
- **OK -** Saves any changes that have not been applied and closes the window.

- **Cancel -** Cancels any changes made to the options. Any actions confirmed with the **Apply** button cannot be restored with **Cancel** but must be entered and confirmed again.

Distributed Exporter

The Distributed Exporter is a utility that creates and provides necessary information to the Extension Technology so that actual subsetting of your distributed data store can occur.

This utility generates the following files after you have successfully exported from the Data Subset Extraction module:

- `method.rc` - Coded content of source and target databases information.
- `method.txt` - Information for the environment and the method in it which are about to be masked.
- `filter.txt` - All created filters in the test environment.
- `elab.txt` - Data that is about to be masked.
- `cbfld.txt` - Information about combined fields used in the method.
- `CREATETABLE.sql` - list of the tables that need to be masked; a generic CREATE TABLE statement is provided for each table.
- `CREATEINDEX.sql` - list of all the indexes for the tables that need to be masked; a generic CREATE INDEX statement is provided for each index.
- `ALTERTABLE_RI.sql` - list of all the primary and foreign keys for the tables that need to be masked; a generic ALTER TABLE statement is provided for appropriate primary and foreign keys.

There are two areas of interest when it comes to using the Distributed Exporter with distributed data stores.

- If you use ODBC-enabled data stores, you can subset across all your data stores with one invocation of the ODBC Extension. This feature is not available with the Oracle Extension.
- You also have the ability to subset within one distributed data store, as long as a distinct target schema name is provided.

Note: If you use the Distributed Exporter utility on one machine, and do your subsetting on another, make sure that the target ODBC DSN matches the ODBC DSN used for the actual subsetting. Similarly, if you use the Oracle Extension, make sure your Oracle TNS service names match.

To access the **Distributed Exporter** window, click from the main window.

Window contents

This section describes the fields listed in the **Distributed Exporter** window.

To export a method from the Data Subset Extraction module, you need to provide the following input:

- Your workspace (machine ID, company).
- Your source data store credentials (source database, source schema, source user ID/password).
  - For a data store that you will access using the ODBC Extension, the source database is an ODBC DSN configured to access that data store.
  - For a data store that you will access using the Oracle Extension, the source database is a BDE alias configured to access that data store. For more information, see the *Oracle Alias Creation* section in *Getting Started with Distributed Data Stores*.
- Your target data store credentials (target database, target scheme, target user ID/password).
  - For a data store that you will access using the ODBC Extension, the target database is an ODBC DSN configured to access that data store.
  - For a data store that you will access using the Oracle Extension, the target database is an Oracle TNS service name configured to access that data store. For more information, see the *Oracle Alias Creation* section in *Getting Started with Distributed Data Stores*.

- Directory where you want to place output files for Extension Technology (Output directory).

Note: If the source database and the target database are the same, the schemas cannot be the same.

In addition, if you are using ODBC-enabled data stores, you can provide multiple sets of source and target data store credentials.

Available operations

The buttons at the bottom of the **Distributed Exporter** window are:

- **Export** - Starts the export procedure, which generates the files.
- **Close** - Closes the window.

Distributed Statistics Loader

For the Data Express for Distributed Data Stores solution, the Distributed Statistics Loader is a utility that lets you load run-time statistics into the KB. Once you have executed the Extension Technology to subset your data, a log filed called `dxe_statistics.txt` is created. This file is used to load the run-time data into the KB so that the statistics are available in Data Subset Extraction.

For more information on using the Distributed Statistics Loader, see *Extension Technology Utilities* in *Getting Started with Distributed Data Stores*.

To access the **Distributed Statistics Loader** window, click from the main window .

Window contents

This section describes the fields listed in the **Distributed Statistics Loader** window as shown below:

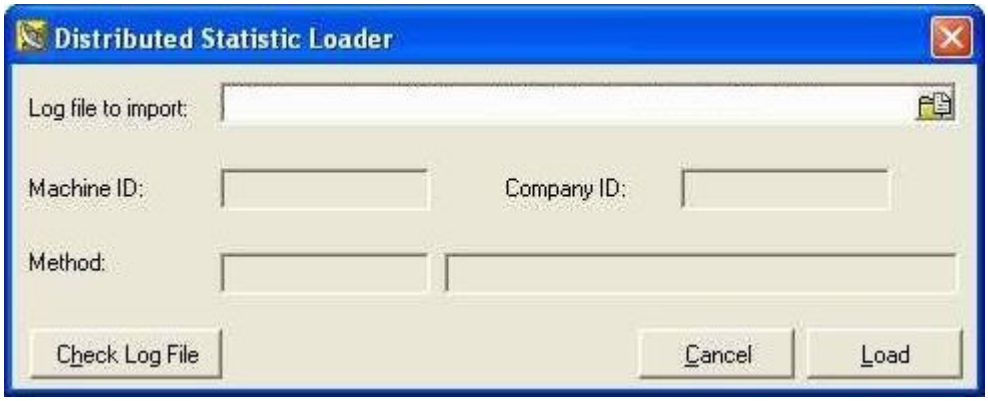

Figure 8: Figure 16-1: Distributed Statistic Loader

To use the Distributed Statistics Loader:

1. In the Log file to import box, specify the `dxe_statistcs.txt` log file name including the directory path where you saved it on your Windows machine.
2. Click **Check Log File** to verify that the log file you select is correct.

   The remaining fields in the **Distributed Statistic Loader** windows populate from information specified from the `dxe_statistcs.txt` file.
3. Click **Load** to load the subset extraction statistics into Data Subset Extraction.

   A `Statistic loaded successfully!` message is displayed.

You can view the run-time statistics in various places in Data Subset Extraction. For example:

- Viewing the List of Elaborations in Step by selecting the appropriate method from the Main Window
- Viewing the Data Stores Elaboration from the Elaborations tab in the Work with Method window
- Viewing the List of Elaborations in Step from the Method Overview tab in the Work with Method window

Run-time statistics include the following:

- Number of Read Records
- Number of Written Records
- Reduction in Percentage
- Simulation Date
- Real Extraction Date
- Elapsed Time [hh:mm:ss]
- CPU Time [hh:mm:ss]

Note: The Reduction in Bytes run-time statistic is not used in the Distributed Systems solution because of the way ODBC-enabled data stores display the record length.

Available operations

The buttons at the bottom of the **Distributed Statistics Loader** window are:

- **Check Log File** - Verifies that the log file you selected is correct.
- **Cancel** - Closes the window.
- **Load** - Loads the subset extraction statistics into Data Subset Extraction.

Work with Method multiple filters

In order to create multiple filters adding conditions like "OR" or "AND" to a sub-extraction query, you can apply a new class with a filter to a second column of a data store already included in any method or you can create a combined data element with fields you need to use and include it in a method.

Introduction to Work with Method multiple filters

When more than one filter (single or combined) is added to the same data store, the sub-extraction query is created with UNION or INTERSECT SQL statements. This enables extraction of records that satisfy the filter criteria.

To better describe the two different situations, this guide divides the explanation in two different case:

- More than one single filter for the same data store. This filter adds an "OR" or an "AND" condition to the query.
- More than one combined filter for the same data store, involving more than one data element. This filter adds an "OR" or an "AND" condition to the query

For both cases, we used a DB2 table named CUSTOMER, with columns NAME and SURNAME and related class NAME and SURNAME.

More than one single filter for the same data store

This following steps describe how to apply the second filtered class SURNAME to the second column SURNAME of a data store CUSTOMER which has been included in a Method from the **Work with Method** window. The query is created with a UNION statement within the two filter criteria.

At the end of the Method confirmation phase, in order to apply the second class, copy the data store to a new step by clicking **Copy Selected Step**. To copy the data store:

1. Select the appropriate row in the **Work with Method** window and click the button.
2. Right-click on new row to display the context menu, point to **Elaboration**, and then click **Clear Selection class** and **Disconnect Filter**.
3. Right-click on new row to display the context menu, point to **Elaboration**, and then click **Selection class / Filter properties** and select the class SURNAME assigned to column SURNAME.
4. Right-click on new row to display the context menu, point to **Move selected to** and then click **20 - STEP CLASSES**.

To create a query with the INTERSECT statement, follow steps **1** through **4** above, and then:

1. Select the two rows related to the DB2 table CUSTOMER.
2. Right-click; then select **Utility** from the context menu.
3. Select the **Create filter in AND** voice.

Filter types

This section describes the filter types that can be applied to the primary and secondary classes listed in the List of class in the elaboration grid in the **Selection class / Filter properties** window.

If you want to apply a filter to a primary and secondary class, you must select each class from the list and later select the type of filter you want to specify. This operation must be repeated for the secondary class in the list for which you want to apply a filter.

Note: The filter values for date fields must be in the format *YYYYMMDD* or *YYYYMMDD HHHH.MM.SS* where *HHHH* is based on a 24-hour clock.

Note: After associating the correct filter with each class, click **OK**.

Filter examples more than one single filter

The following examples show various ways to code for more than one single filter:

Example: Filter by range in OR (UNION)

The following example reproduces the query created using filter by range value for NAME and SURNAME , where the values are:

- from ALBERT to STEPHEN for NAME
- from LENNON to MORANIS for SURNAME

```
SELECT A.* FROM "DELGT"."CUSTOMER" A
WHERE A.NAME BETWEEN 'ALBERT' AND 'STEPHEN'
UNION
SELECT A.* FROM DELGT.CUSTOMER A
WHERE A.SURNAME BETWEEN 'LENNON' AND 'MORANIS'
```

Example: Filter by range in AND (INTERSECT)

The following example reproduces the query created using filter by range value for NAME and SURNAME , where the values are:

- from ALBERT to STEPHEN for NAME
- from LENNON to MORANIS for SURNAME

```
SELECT A.* FROM "DELGT"."CUSTOMER" A
WHERE A.NAME BETWEEN 'ALBERT' AND 'STEPHEN'
INTERSECT
SELECT A.* FROM DELGT.CUSTOMER A
WHERE A.SURNAME BETWEEN 'LENNON' AND 'MORANIS'
```

Example: Filter by list value in OR (UNION)

The following example reproduces the query created using filter by list value for NAME and SURNAME , where the values are:

- ALBERT, TOM and STEPHEN for NAME
- LENNON, ROSS and MORANIS for SURNAME

```
SELECT A.* FROM "DELGT"."CUSTOMER" A
WHERE ((A.NAME) IN ('ALBERT','TOM' ,'STEPHEN'))
UNION
SELECT A.* FROM DELGT.CUSTOMER A
WHERE ((A.SURNAME) IN ('LENNON', 'ROSS', 'MORANIS'))
```

Example: Filter by list value in AND (INTERSECT)

The following example reproduces the query created using filter by list value for NAME and SURNAME , where the values are:

- ALBERT, TOM and STEPHEN for NAME
- LENNON, ROSS and MORANIS for SURNAME

```
SELECT A.* FROM "DELGT"."CUSTOMER" A
WHERE ((A.NAME) IN ('ALBERT','TOM' ,'STEPHEN'))
INTERSECT
SELECT A.* FROM DELGT.CUSTOMER A
WHERE ((A.SURNAME) IN ('LENNON', 'ROSS', 'MORANIS'))
```

More than one combined filter involving more than one data element

This following section describes how to apply the filtered class CLA0202 assigned to the combined data element which has been created including the column NAME and SURNAME of data store CUSTOMER, from the **Work with Method** window.

Filter types

This section describes the filter types that can be applied to the primary classes listed in the List of class in elaboration grid in the **Selection class / Filter properties** window.

If you want to apply a filter to a class and there are several primary selection classes, you must select each class from the list and later select the type of filter you want to specify. This operation must be repeated for each class in the list for which you want to apply a filter.

Note: The filter values for date fields must be in the format *YYYYMMDD* or *YYYYMMDD HHHH.MM.SS* where *HHHH* is based on a 24-hour clock.

Note: After associating the correct filter with each class, click **OK**.

Filter examples more than one data element

Example: Filter by range

The following example reproduces the query created using filter by list value for `NAME` and `SURNAME`, where the values are:

- ALBERT,TOM and STEPHEN for NAME
- LENNON, ROSS and MORANIS for SURNAME

```
SELECT * FROM "DELGT"."CUSTOMER" A, OWUR41SV.HSENVPAR B WHERE B.METHOD =
'DOUBCOND3 ' AND B.MCRECID = 0028 AND B.CLATYPE = 0202 AND B.FLTPRG = 0001 AND
A."NAME" = ( ( SUBSTR(B.VALUE, 001, 040) ) ) AND A."SURNAME" =
( ( SUBSTR(B.VALUE, 041, 040) ) ) AND B.PARMPRG > 2
```

Example: Filter by list value

The following example reproduces the query created using filter by list value for `NAME` and `SURNAME`, where the values are:

- ALBERT,TOM and STEPHEN for NAME
- LENNON, ROSS and MORANIS for SURNAME

```
SELECT * FROM "DELGT"."CUSTOMER" A, OWUR41SV.HSENVPAR B WHERE B.METHOD =
'DOUBCOND3 ' AND B.MCRECID = 0028 AND B.CLATYPE = 0202 AND B.FLTPRG = 0001 AND
A."NAME" = ( ( SUBSTR(B.VALUE, 001, 040) ) ) AND A."SURNAME" =
( ( SUBSTR(B.VALUE, 041, 040) ) ) AND B.PARMPRG > 2
```

Example: Filter by filter list

. The following example reproduces the query created using filter by filtered where class `NAME` and class `SURNAME` have been previously created like output class for another entry within the same method:

```
SELECT * FROM "DELGT"."CUSTOMER" A, OWUR41SV.ANENVLST B WHERE B.METHOD =
'DOUBCOND5 ' AND B.MCRECID = 0028 AND B.CLATYPE = 0202 AND B.FLTPRG = 0001 AND
A."NAME" = ( ( SUBSTR(B.VALUE, 001, 040) ) ) AND A."SURNAME" =
( ( SUBSTR(B.VALUE, 041, 040) ) )
```