MICRO FOCUS

Data Express 4.0

Data Generation Guide

Contents

Data Generation Guide

This guide outlines the concepts and procedures used by the Data Express Data Generation module, which uses user-specified selection criteria, rules, and data to create a new test environment from scratch.

⚠ **Important:** Data Generation is available for ODBC distributed data stores only. For more information, see the *Front End Guide*.

You begin with an existing but empty test environment. Then, use the Data Generation module to:

- Create rules that derive a test environment from the client side
- Populate job submissions derived from the client side
- Load tables via the Distributed Loader
- Define and associate selection classes

Use this process to generate a full test environment, or a partial test environment. A partial test environment typically includes the applications not present in production. In this case, the remainder of the test environment comes from production through Data Subset Extraction.

Who Should Read This Guide

This guide is for Micro Focus users who are interested in using a subset of data to populate empty tables independent of the content of a production environment. It explains the Data Generation configuration procedure, and describes the steps required to populate the files in your application.

Before using this product, we recommend that you carefully review the other Data Express documentation, including the *Front End Guide* and the *Data Subset Extraction Guide* in particular.

Before Starting with Data Generation

Before using the Data Generation module for Data Express bear in mind the following points:

- The Data Generation module is available in Data Express for Distributed Systems only. This requires that you have the correct license. See the *Installation Guide* for details.
- To use the Data Generation module, it is not necessary for you to analyze application or program source code.
- The Data Generation module manages all types of ODBC distributed data stores processed by the Data Builder module.

Getting Started

Provides an overview of the Data Generation module. We recommend that you start by reading this section before using the module.

Mode

The Data Generation module is available for Distributed Systems only, and therefore must be used in Standard mode. This means direct access to the data resident on an RDBMS from your PC.

For Standard mode configuration information and instructions, see the *Appendix A. Standard Mode* section in the *Front End Guide*.

Data Generation UI Guidelines

The following guidelines apply to windows and screen components of the Data Generation module:

- The main window, **Data Generation - [Master Form]**, cannot be closed except when exiting the module.
- Multiple MDI windows may be open at the same time.
- To view a list of commands that apply to a specific item or area, right-click on the item or area.

Starting the Data Generation Module

To start the Data Generation module:

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

   Note: If you are using Data Express on Windows Vista and User Access Control is enabled, you must run Data Generation as an 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 Generation module.

   Note: To connect to the database for first time, you need your user ID and password.

3. Select the required schema. The **Data Generation - [Master Form]** window appears:
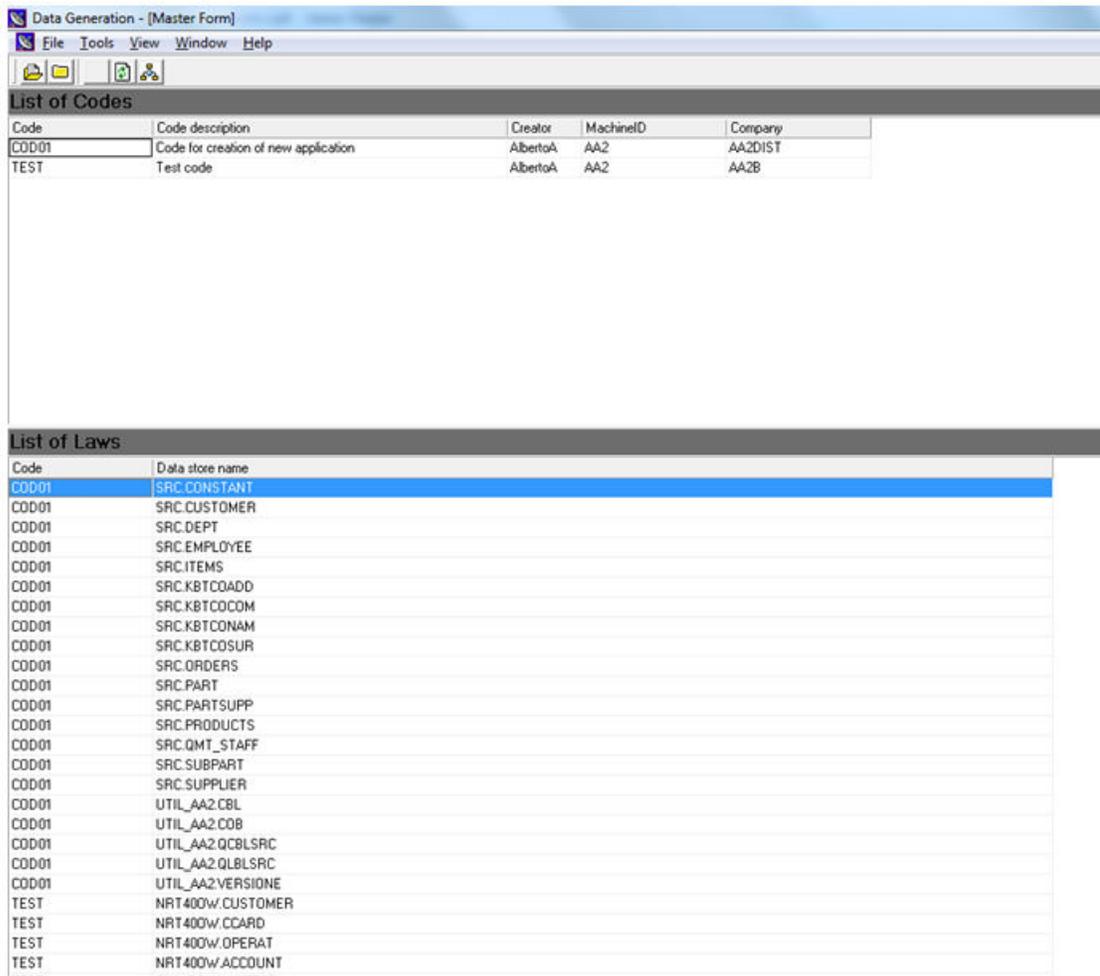

Figure 1: Data Generation - [Master Form] Window

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

Directive Types

The following directives, applied at column level, enable the generation of synthetic data according to their defined rules. If no directive is specified for a given column, the column is populated with the default value for column type; for example, zero for numeric columns:

Generation by dictionary	Column values are obtained from a dictionary. Data Express provides some pre-loaded dictionaries (for masking purposes), and allows you to add dictionaries as well. To generate using a dictionary, you specify the dictionary name in the directive properties.
Generation by domain value list	Column values are obtained from a user-defined list. For example,if your column is a CHAR(1) and you want to assign it the values M and F (as male and female) with probability distribution values to represent 60% M and 40% F, you can specify these values in the directive properties.
Generation by random value list	Columns are randomly assigned values in the column's domain of definition.
Generation by referential integrity (other file dependency)	Column values are obtained from a list of values present in a column from another table. To ensure this, the parent and child column must be of the same class. **Note:** This feature is similar to the "filter by filtered list" feature in Data Subset Extraction.

Distributed Exporter Utility

The Distributed Exporter utility that creates and provides necessary information to the Extension Technology that enables the generation of a distributed data store.

This utility generates the following files after you have successfully exported from the Data Generation 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`	Information about tables involved in Data Generation process.
`Dir.txt`	Information about the directives to be used
`cbfld.txt`	Information about combined fields used in the method.
`CREATETABLE.sql`	A list of the tables that need to be masked; a generic CREATE TABLE statement is provided for each table.
`CREATEINDEX.sql`	A 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`	A 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.

If you use the Distributed Exporter utility on one machine, and do your generation on another, make sure that the target ODBC DSN matches the ODBC DSN used for the actual data generation.

Executing Data Generation

Data Generation execution is done from the command line using the `dxestart` command, which reads the configuration files exported by Distributed Exporter, and populates the target environment according to the laws and directives specified.

The `dxestart` executable resides in the `C:\Program Files (x86)\Micro Focus\Data Express 4.0\Synthetic\ODBC` directory by default.

For complete information about using the `dxestart` command, see *The dxestart Command* topic in your *Getting Started with Distributed Data Stores* documentation.

Catalog Data

Data Generation supports only distributed data stores, cataloged via ODBC. Therefore, you must catalog the target database in Data Express by using the Distributed Loader feature of the Data Builder module. A detailed description of this feature is in the *Using Distributed Loader* section of your *Getting Started with Distributed Data Stores* documentation. Consider the following:

- When using Distributed Loader, the target database should be accessible via ODBC using the `dxegenloadfile` utility either from the machine where the Data Builder module is installed or from the machine where the distributed engine is installed. The `dxegenloadfile` utility is detailed in your *Getting Started with Distributed Data Stores* documentation.
- Distributed Loader allows you to specify an ODBC instance and schema, and presents a list of tables from which you select tables to load into the Data Express knowledge base.
- After confirming which tables to load, Distributed Loader launches an interactive process that writes the catalog and structure information for the selected tables to the knowledge base.

Classify Data

The Data Generation module requires that you assign a class to each column that uses referential integrity directives. For instructions on assigning classes, see the *Importing classes* section in your *Getting Started with Distributed Data Stores* documentation.

> **Note:** While you can assign classes to columns that are not used in Data Generation, it is not required.

Define Rules

Use the Create Code wizard to define Data Generation rules. The Create Code page of the wizard enables you to create, name, and describe a code set for a given machine and company:
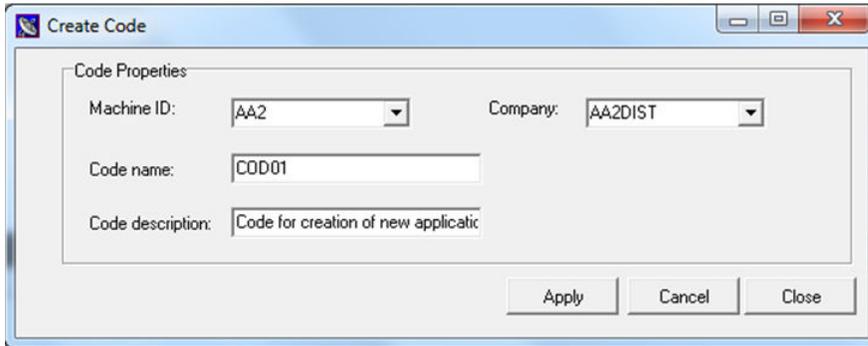
After applying the parameters specified on the Create Code page, you then progress to the Work With Directives page. On this page, you associate the code set with tables by adding a law set rule for each table. Each law rule definition includes a parameter that specifies the number of records used to populate the associated table.
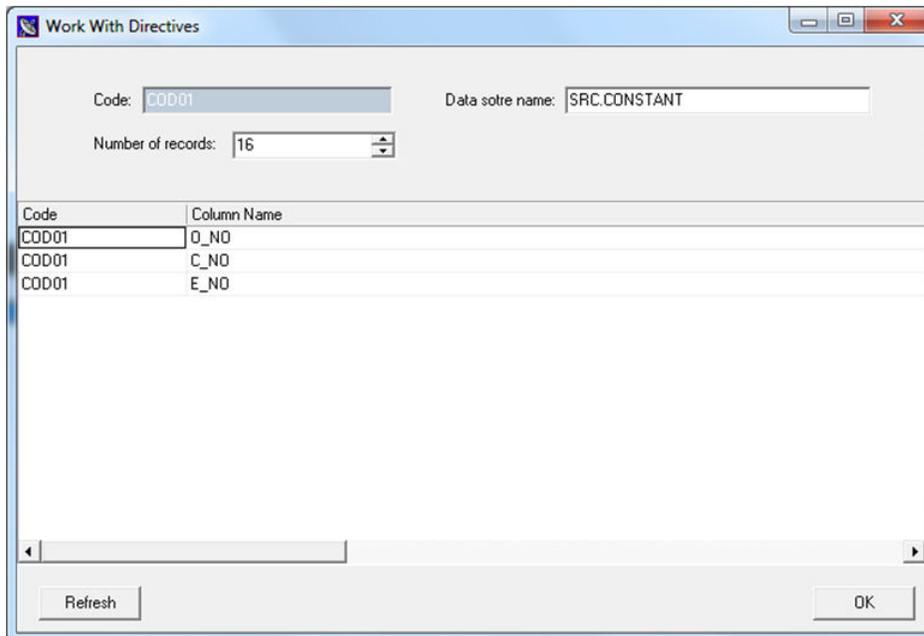

From this page, you can open each law rule, enabling you to specify directives when necessary. To do this, right-click a **Column Name**; then select **Set Directive**. This opens the Create a Directive for column page.

You can select the directive type, and assign a routine name to the directive type. Each directive type obtains column values differently as follows:

Other file dependency   These directives obtain all relevant information from the class assignment. The values for the new column are obtained from the column in the parent table that is associated with the same class.

Create a Directive for column, Other file dependency
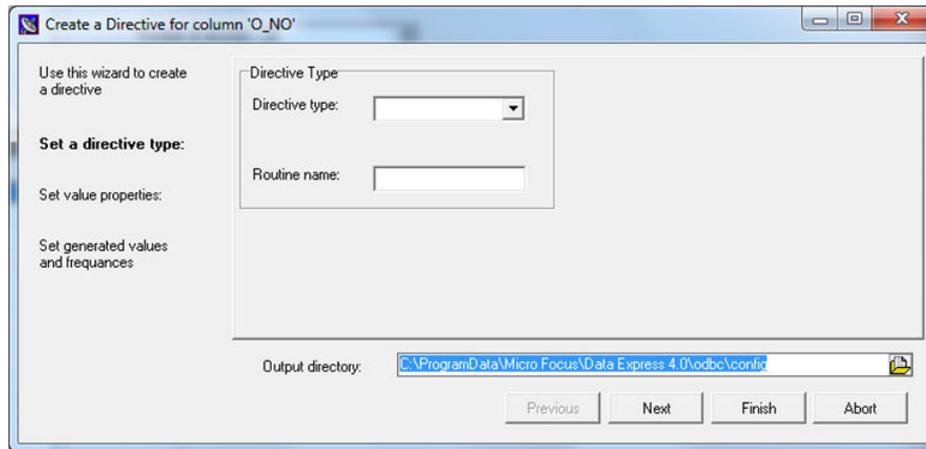
Dictionary   These directives require that you specify the dictionary name. For further information on Data Express dictionaries, see the *Masking Routines* section in your *Getting Started with Distributed Data Stores* documentation.

Create a Directive for column, Dictionary
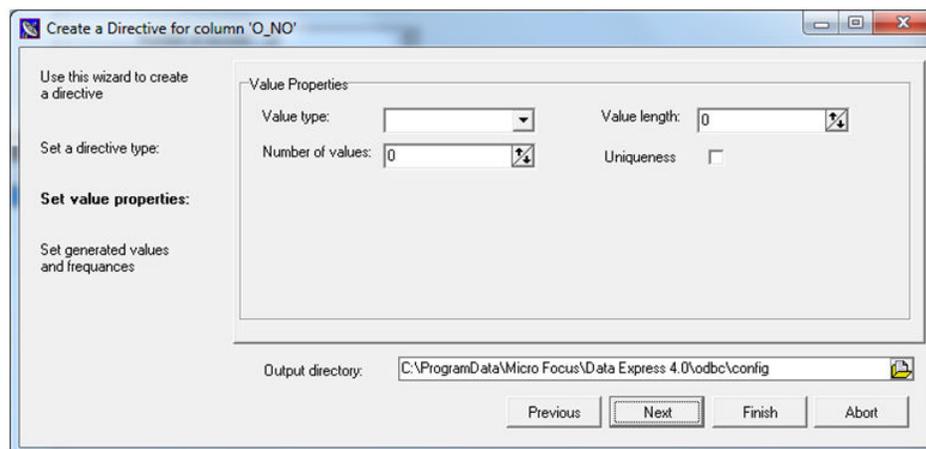
Domain value list   These directives require that you specify the different values the column can have, and a frequency for each value.

⚠ **Important:** Columns with no directives associated receive the default value for the associated column type.

Export Rules

Use the Export wizard to export rules at the code set level. The wizard enables you to choose the machine, company, and code, and to specify the output directory. The output directory should be the folder that contains the configuration files.
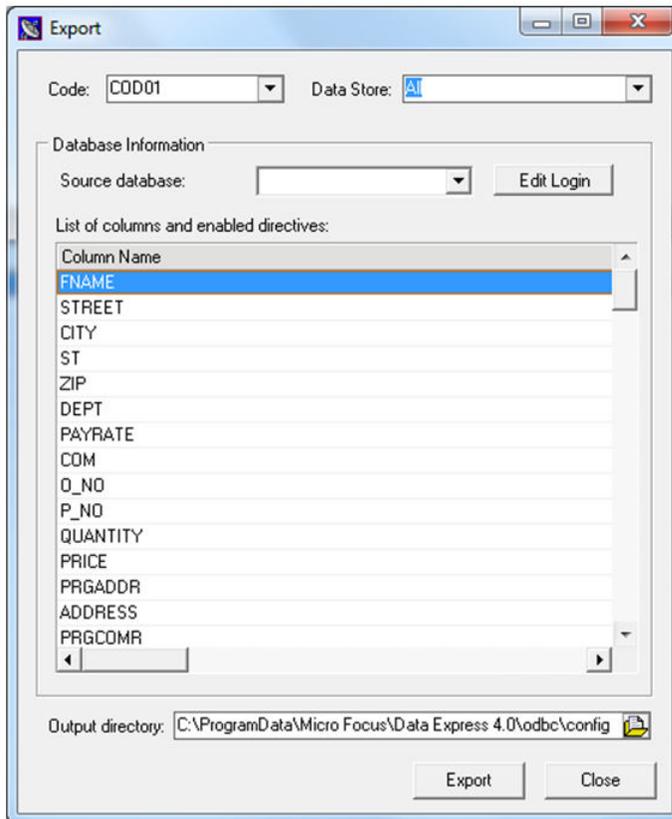
Note: The **List of columns and enabled directives** is on the Export page as a summary, and is not editable.

Execute Rules

To execute Data Generation rules, you must have a run-time knowledge base. A run-time knowledge base is tables in the same database instance that contains the target tables to be populated, or in a separate instance if you use a centralized run-time knowledge base. Details about run-time knowledge base creation, including a centralized run-time knowledge base, are described in the DBA Tasks for Extension Technology section of your *Getting Started with Distributed Platforms* documentation.

Once the run-time knowledge base is in place, the execution of rules is done by executing the `dxestart` command. See *The dxestart Command* topic in your *Getting Started with Distributed Platforms* documentation.

Index

A

Audience 4