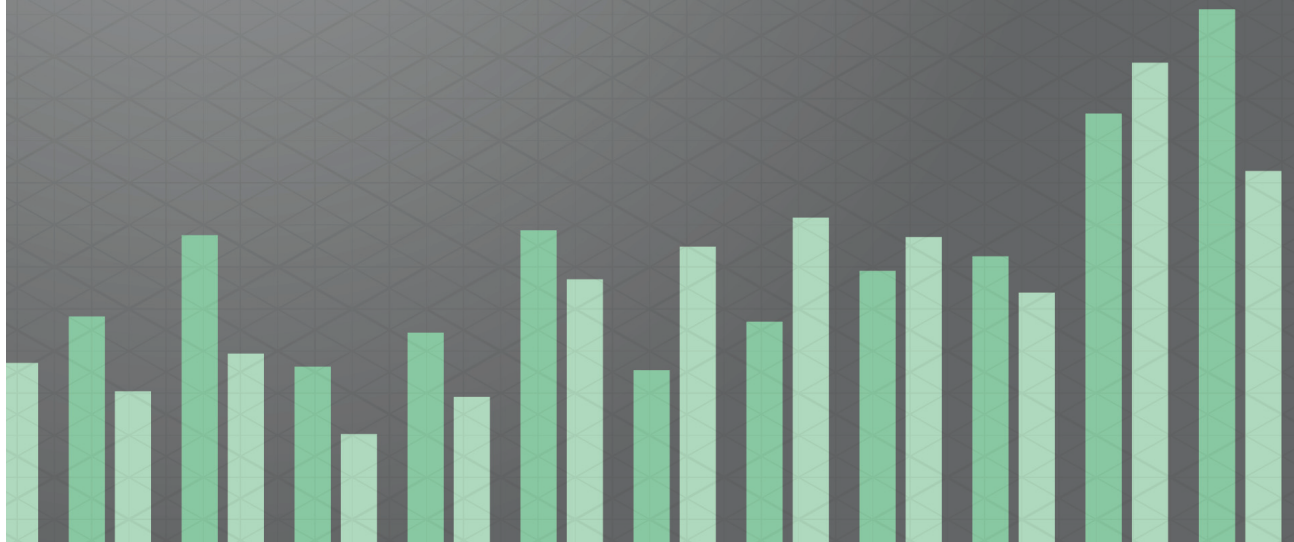




ACTUATE.
The BIRT Company™



BIRT Analytics



Using BIRT Analytics Loader

Information in this document is subject to change without notice. Examples provided are fictitious. No part of this document may be reproduced or transmitted in any form, or by any means, electronic or mechanical, for any purpose, in whole or in part, without the express written permission of Actuate Corporation.

© 2003 - 2014 by Actuate Corporation. All rights reserved. Printed in the United States of America.

Contains information proprietary to:
Actuate Corporation, 951 Mariners Island Boulevard, San Mateo, CA 94404

www.actuate.com

The software described in this manual is provided by Actuate Corporation under an Actuate License agreement. The software may be used only in accordance with the terms of the agreement. Actuate software products are protected by U.S. and International patents and patents pending. For a current list of patents, please see <http://www.actuate.com/patents>.

Actuate Corporation trademarks and registered trademarks include:

Actuate, ActuateOne, the Actuate logo, Archived Data Analytics, BIRT, BIRT 360, BIRT Analytics, BIRT Data Analyzer, BIRT Performance Analytics, Collaborative Reporting Architecture, Dynamic Data Web, e.Analysis, e.Report, e.Reporting, e.Spreadsheet, Encyclopedia, Interactive Viewing, OnPerformance, Performancesoft, Performancesoft Track, Performancesoft Views, Quite4Me, Quiterian, Report Encyclopedia, Reportlet, The people behind BIRT, X2BIRT, and XML reports.

Actuate products may contain third-party products or technologies. Third-party trademarks or registered trademarks of their respective owners, companies, or organizations include:

Mark Adler and Jean-loup Gailly (www.zlib.net): zlib. Apache Software Foundation (www.apache.org): Axis2, log4, Tomcat. Boost.org: Boost libraries, licensed under the Boost Software License. CURL (curl.haxx.se): Curl, licensed under a MIT/X derivate license. International Components for Unicode (ICU): ICU library. Marcin Kalicinski (rapidxml.sourceforge.net): RapidXML, licensed under the Boost Software License. Bruno Lowagie and Paulo Soares: iTextSharp, licensed under the Mozilla Public License (MPL). Math.NET: Math.NET, licensed under the MIT/X11 License. Microsoft Corporation: Access Database Engine, SQL Server Express. openssl team (sourceforge.net): openssl.org: OpenSSL, licensed under the OpenSSL license. gooxdoo.org: gooxdoo, licensed under the Eclipse Public License (EPL). Dave Scriven (svg.codeplex.com): SVG Rendering Engine, licensed under the Microsoft Public License. SQLAPI: SQLAPI++. sqlite.org: SQLite, public domain. stlsoft.org: STLSoft libraries, licensed under the BSD license. Matthew Wilson and Garth Lancaster (www.pantheios.org): Pantheios, licensed under a modified BSD license.

All other brand or product names are trademarks or registered trademarks of their respective owners, companies, or organizations.

Document No. 131215-2-580321 February 19, 2014

Contents

About <i>Using BIRT Analytics Loader</i>	v
Chapter 1	
Introducing BIRT Analytics Loader	1
About BIRT Analytics Loader	2
Understanding BIRT Analytics Loader	2
Understanding BIRT Analytics users	3
Getting started	3
Chapter 2	
Working with projects	5
About BIRT Analytics Loader projects	6
Planning a project	6
Understanding the FastDB repository	7
Creating a new project	7
Managing projects	9
Laying out the feature tabs	10
Running a load process	10
Stopping the IIS service	10
Stopping IIS using Services	11
Stopping IIS using a command-line editor	11
Running a project	12
Using reports	13
Synchronizing data	13
Synchronizing data using BIRT Analytics Admin	13
Synchronizing data using bashell	17
Managing FastDB Engine accounts	18
Chapter 3	
Extracting data	19
About extracting data	20
Connecting to data	20
Connecting to databases	20
Using native drivers	21
Using ODBC drivers	23
Connecting to files	24
Uploading local files	25
Uploading remote files	26
Uploading files using a file list	27
Updating data source definitions	28
Creating a load process	28
About transactions	28
Setting load process options	29
Creating a load transaction	29
Mapping data from files	30
Defining a target database	30
Defining a source file format	31
Defining table columns	33
Mapping data from databases	37
Defining a data set	37

Chapter 4	
Managing data	41
About managing data	42
Understanding Explorer	42
Refreshing the repository content	43
Exploring table properties	43
Exploring column properties	43
Dropping database objects	45
Understanding dynamic and permanent columns	45
Understanding Links	46
Chapter 5	
Transforming data	49
About transforming data	50
Understanding Includes	50
Understanding scripts	51
Understanding Repository	52
Using the text editor	53
Creating new columns	53
AGGREGATE	53
DECODE	55
DOMAIN	57
EXPRESSION	60
INUMERIC	65
PARAMETRIC	67
QUANTILE	69
RANKING	71
Managing data objects	73
DROPCOLUMN	73
DROPDATABASE	73
DROPTABLE	74
DROPLINK	74
LINK	75
MAKEPERMANENT	76
RENAME	77
Miscellaneous instructions	77
EXPORT	78
IFEXISTS	81
IFLINKED	82
INCLUDE	83
PRINT	84
VAR	85
Chapter 6	
Administering BIRT Loader	87
About BIRT Loader administration	88
Managing users	88
Scheduling load processes	89
Using External call	91
Running projects using qLauncher	92
Understanding the BIRT Loader internal database	94
Copying a load project to a new location	95
Managing the repository	96

Creating a new database repository	97
Copying a data repository to a new location	98
Changing the repository content using the same location	99
Adding a BIRT Analytics database to a repository	99
Glossary.....	101
Index	115

About Using BIRT Analytics Loader

BIRT Analytics is a web application, data repository, and set of web services that support analysis and forecasting of big data.

BIRT Analytics consists of three key components:

- Actuate BIRT Analytics, a tool that is used to carry out dynamic analyses
- BIRT Analytics Administration, a set of tools that supports administering access and privileges for each module and component of the BIRT Analytics product
- BIRT Analytics Loader, a tool that extracts, transforms, and loads records from an external data source to FastDB, the BIRT Analytics data repository

Using BIRT Analytics Loader describes how to use Actuate BIRT Loader to collect data from various sources, transform the data depending on business needs, and load the data into a destination database.

This document is a guide for ETL (Extract, Transform, and Load) developers who understand basic concepts in working with data and are familiar with the data structure. Technical concepts and explanations about how to accomplish common activities are included in the following chapters. For more detailed information about any of the subjects discussed, see the complete set of Actuate documentation included with Actuate software or contact your database administrator.

Using BIRT Analytics Loader includes the following chapters:

- *About Using BIRT Analytics Loader.* This chapter provides an overview of this guide.
- *Chapter 1. Introducing BIRT Analytics Loader.* This chapter describes how to access and log in to BIRT Analytics Loader.
- *Chapter 2. Working with projects.* This chapter describes how to create and run projects to load data.
- *Chapter 3. Extracting data.* This chapter describes how to create data connections and data mappings.
- *Chapter 4. Managing data.* This chapter explores the data elements, such as databases, tables, and columns, and their properties.
- *Chapter 5. Transforming data.* This chapter describes how to transform data.
- *Chapter 6. Administering BIRT Loader.* This chapter describes how to manage users and schedule data load processes.
- *Glossary.* This chapter provides definitions of terms used in the BIRT Analytics product and documentation.

1

Introducing BIRT Analytics Loader

This chapter contains:

- About BIRT Analytics Loader
- Understanding BIRT Analytics Loader
- Getting started

About BIRT Analytics Loader

Business analytics depend on sufficient volumes of high-quality data. The difficulty in ensuring data quality is integrating and reconciling data across different systems, and then deciding what subsets of data to make available to the analysis tool.

BIRT Analytics Loader provides one, integrated tool for extracting, transforming, and loading (ETL) big data into BIRT Analytics. ETL pulls data from one, or multiple data sources and places it into another database, using the following three processes:

- Extract reads data from a database.
- Transform converts the extracted data from its previous form into the form required for placement in another database. The transformation process uses rules, or lookup tables, or combines the data with other data.
- Load writes the data into the target database.

Use BIRT Analytics Loader to load big data for analysis. Use BIRT Analytics to analyze the data. Both tools run in a web browser, and ship as options in the Actuate BIRT Analytics product.

Understanding BIRT Analytics Loader

BIRT Analytics Loader is a web-based application you use to create and execute processes that build databases for data mining and reporting. Using BIRT Loader, you can schedule the load processes to run at convenient times. As the diagram in Figure 1-1 shows, BIRT Analytics Loader is installed on a Tomcat web server.

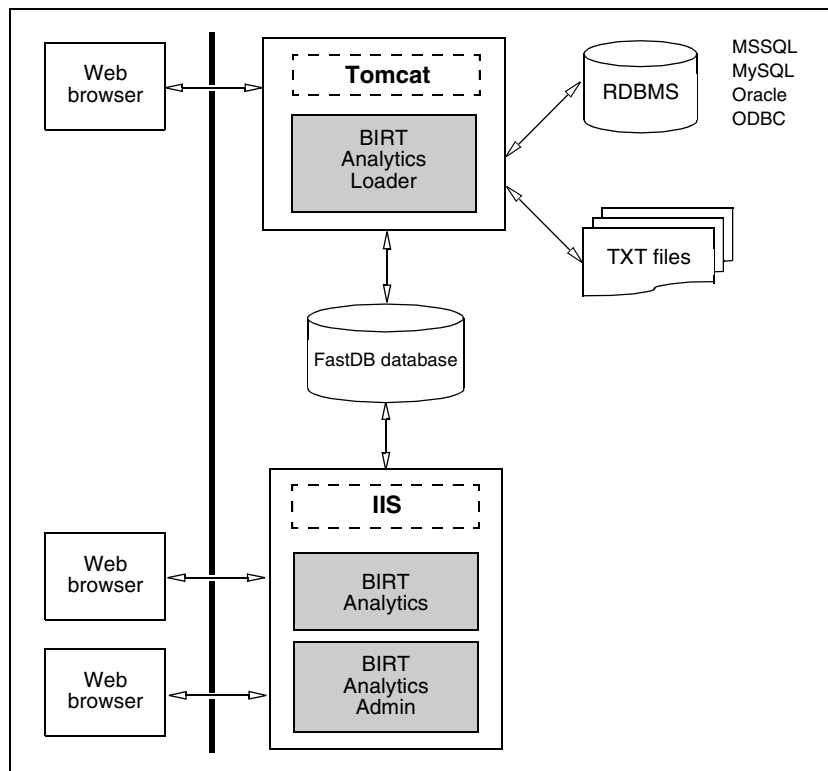


Figure 1-1 Actuate BIRT Analytics architecture

BIRT Analytics Loader supports connections to different data sources, relational databases, and text files, using database native drivers and ODBC connectors. The application builds processes that extract, transform, and load data, storing it in a proprietary format in a FastDB database. Actuate FastDB database is a file system database, managed by the FastDB engine. The engine runs as a service, named Dubnium Service. You can create multiple databases and select which one to be active.

Use BIRT Analytics to analyze and report from this data. Use BIRT Analytics Administration tool to synchronize the data, imported by BIRT Loader, with BIRT Analytics. Both applications, BIRT Analytics and BIRT Analytics Administration, are installed on a web server, Microsoft Internet Information Services.

Understanding BIRT Analytics users

There are four types of users in Actuate BIRT Analytics products:

- **BIRT Analytics users**
Users who log in to BIRT Analytics to do data mining and analysis.
- **BIRT Analytics Administration users**
Users who log in to BIRT Analytics Administration and manage access to objects, specify security filters, define password policies, browse logs, synchronize the database, specify password policies, and perform other security maintenance actions.
- **BIRT Analytics Loader users**
Users who log in to BIRT Analytics Loader to create projects for loading external data into a FastDB analytical database.
- **FastDB Engine users**
The FastDB Engine user is the internal account the Dubnium server uses to communicate with the Actuate FastDB database and the rest of the internal modules. This account provides an additional layer of security for protecting sensitive data. When you create a project for loading data you must provide the secure account used by the engine to access the data. For more information on administering the Dubnium server data security account, see “Managing FastDB Engine accounts” in Chapter 6, “Administering BIRT Loader.”

Getting started

Open BIRT Analytics Loader to create projects that build database loading processes. Actuate BIRT Analytics Loader installs as part of the Actuate BIRT Analytics product. For more information, see the BIRT Analytics installation guide.

To access BIRT Analytics Loader, you need a web browser.

How to log in to BIRT Analytics Loader

- 1 You open the BIRT Analytics Loader web application using a hyperlink or a web address similar to the following URL:

`http://<webserver>:8080/qloader`

where

- `<webserver>:8080` is the web server and TCP port that runs the BIRT Analytics Loader application. Alternatively, you can use an IP address instead of the web server name.
- `qloader` is the context root to the BIRT Analytics Loader application. The context root is set by the web server administrator when installing the application.

Figure 1-2 displays the BIRT Analytics Loader login page.

The image shows the login page for Actuate BIRT Analytics Loader. On the left, the logo "ACTUATE. BIRT ANALYTICS LOADER" is displayed. On the right, there are input fields for "User:" and "Password:", followed by a "Sign in" button. At the bottom, the Actuate logo and "Actuate Corporation © 2013" are visible.

Figure 1-2 Logging in to Actuate BIRT Analytics

2 To log in, provide the following login credentials:

- In User name, type the user name.
- In Password, type the password associated with the user name.

By default, BIRT Analytics Loader provides an administrator user account, SYSTEM, with the password, PASSWORD. This user is stored into the dubnium.db sqlite database. The user can be modified by editing this database. For more information about dubnium.db sqlite database, see “Understanding the BIRT Loader internal database” in Chapter 6, “Administering BIRT Loader.”

3 Choose Sign in. The BIRT Analytics Loader home page associated with your login credentials appears, as shown in Figure 1-3.

The image shows the home page of the BIRT Analytics Loader after a successful login. The page has a header with the Actuate logo and "BIRT ANALYTICS LOADER". On the right, there is a "Logout" button. The main content area is titled "Projects management" and contains a dropdown menu with "Open", "Remove", and "New" buttons. Below this, there is an "Administration" section with an "Admin" button. The Actuate logo and "Actuate Corporation © 2013" are at the bottom.

Figure 1-3 Viewing the BIRT Analytics Loader home page

After logging in to BIRT Analytics Loader, you can start managing projects. You can choose a project to open, remove a project, or create a new one.

An administrator user, such as SYSTEM, sees the Administrator tab. Options to manage user privileges and schedule loading processes appear on Administrator.

How to log out of Actuate BIRT Analytics

Choose Logout to quit a user session of BIRT Analytics Loader.



When working on a project, you can log out by choosing the logout button on the top right corner of the page.

Working with projects

This chapter contains:

- About BIRT Analytics Loader projects
- Planning a project
- Creating a new project
- Managing projects
- Running a load process
- Synchronizing data
- Managing FastDB Engine accounts

About BIRT Analytics Loader projects

BIRT Analytics Loader organizes data loading activities into projects. Projects are logical descriptions of how you will extract, load, and transform the data into the target database. A BIRT Loader non-administrator user works with only the projects he creates. A project that completes a typical Extract, Transform, Load (ETL) process includes completing each of the following activities, as illustrated in Figure 2-1:

- Defining data sources
- Mapping the data
- Defining the data transformation
- Scheduling the load processes to run

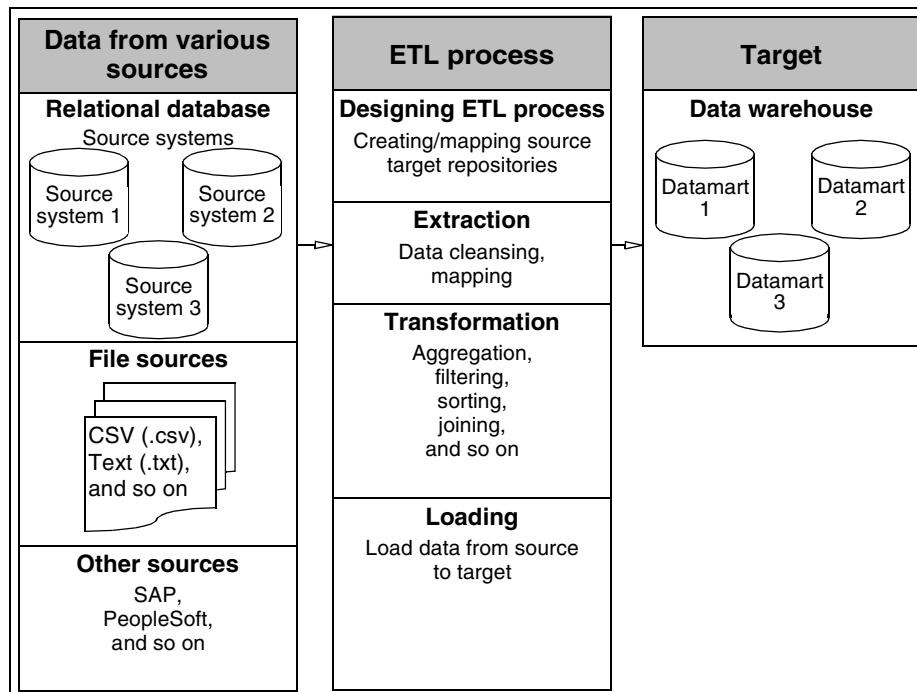


Figure 2-1 Sample ETL process

Planning a project

Typically, there are three key players involved in building an ETL process: a database administrator, a data modeler, and an ETL developer. Sometimes, one person fills multiple roles. The key factor for success when creating an ETL process is to diligently plan the project before starting the ETL process development. Planning a project requires:

- Knowing your data sources and completing a detailed mapping specification. A successful ETL process depends entirely on a completed mapping specification.
- Documenting how you are going to be extracting, cleansing, and transforming your data sources to get to your database table definitions.
- Establishing rules that define how to update database tables. This is an important factor in the ETL design. Depending on the nature of the database, the update can be as simple as a data refresh, cleaning old data, and loading new data. In some occasions, the update could require appending data based on complex criteria such as keyed reads or joins against an

existing dimension before writing the new or updated rows to a database table. Building these rules may potentially add days or weeks to your project timeline.

In the planning process you must define:

- Input and output record layouts
- The location of source and target files and databases
- File and table sizing information
- File and table volume information
- Documentation on how the data will be transformed, if at all

Understanding the FastDB repository

BIRT Analytics Loader loads data into a FastDB database repository, operated by the FastDB Engine. When FastDB is installed, a default data repository is created in this path:

C:\Program files\BIRTAnalytics\dubnium\databases

You can use the same repository to create and store your databases. BIRT Analytics allows you to create a different location for your database repository and set up BIRT Loader to work with this location. For more information on how to create and set up a new data repository, see “Managing the repository” in Chapter 6, “Administering BIRT Loader.”

Creating a new project

When you log in to BIRT Analytics Loader for the first time, there are no existing projects, and the Open and Remove tabs are grayed out, as shown in Figure 2-2.

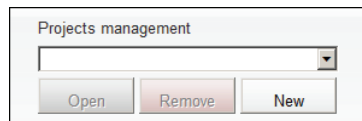


Figure 2-2 Creating a new project

Choose New to create a new project. Create new project appears, as shown in Figure 2-3.

Figure 2-3 Setting a new project

BIRT Loader users can access only projects they create. A BIRT Loader administrator user can access all projects.

How to create a new project

1 To create a new project, you set the following parameters:

- In Name, type the name of the project.
- In IP, type the IP address of the web server where BIRT Analytics Loader is installed.
- In Port, type the port number that the BIRT Analytics Loader application uses. The default port number is 8105.
- Choose SSL if a secure connection is used. In this case the default port number is 8106.
- In User, type the FastDB Engine user. The default user is SYSTEM.
- In Password, type the FastDB Engine user's password. The default password is PASSWORD.

Figure 2-4 displays the settings of a project named MyFirstProject.

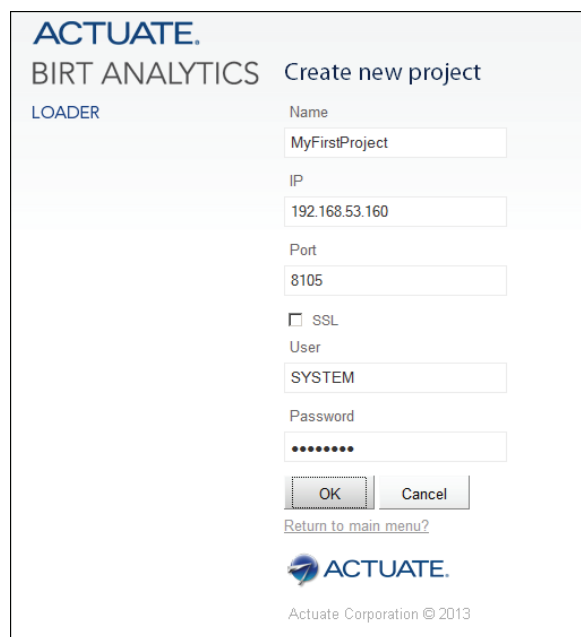


Figure 2-4 Setting up MyFirstProject

2 Choose OK to create the project. The Projects management page appears and the project name is in the drop-down list, as shown in Figure 2-5.

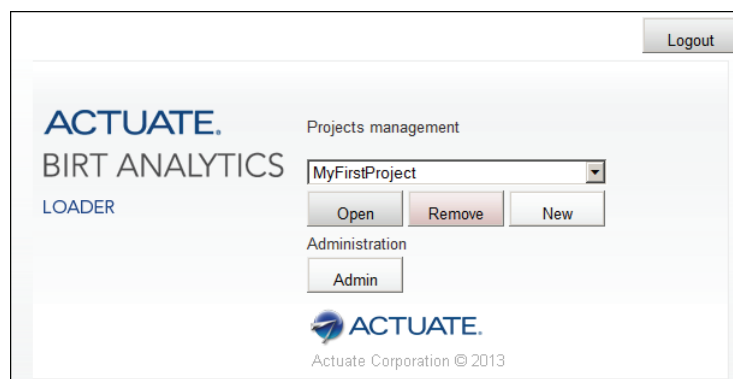


Figure 2-5 Opening an existing project

Managing projects

After opening a project using BIRT Analytics Loader, the project development page appears, as shown in Figure 2-6. The project page contains a top banner, a control menu, feature tabs, and work area. The top banner displays the project name and the logout link.

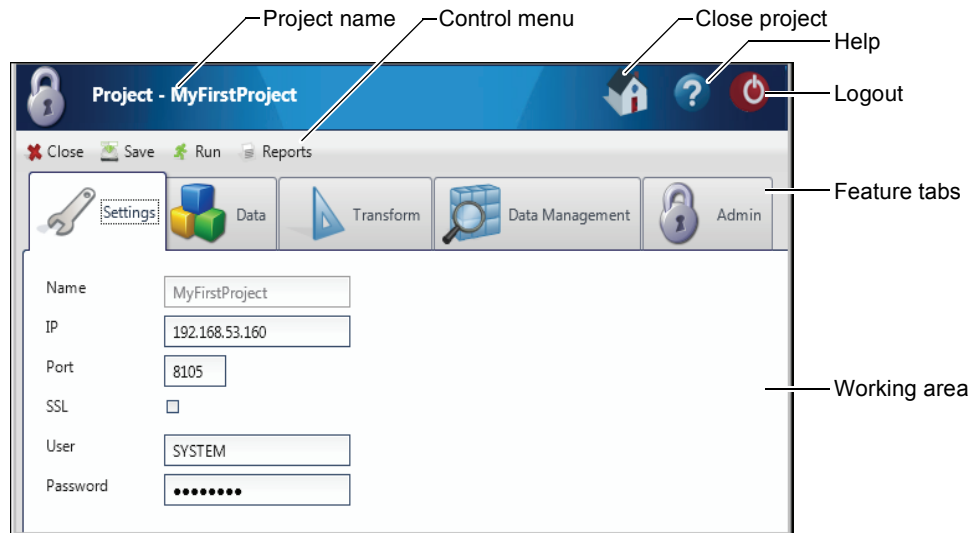


Figure 2-6 Developing a project

The control menu contains the following links, as shown in Table 2-1, used for managing a project.

Table 2-1 Icons for the control menu

Icon	Label	Purpose
	Close	Closes the current project, after confirmation, and returns to the Projects management window.
	Save	Saves the project. To make any change effective, it is necessary to save data by choosing Save.
	Run	Executes the load process.
	Reports	Shows the load process log files.



The feature tabs contain links to the main project activities described in Table 2-2.

Table 2-2 Icons for the feature tabs

Icon	Label	Purpose
	Settings	This tool allows changing the configuration parameters of the current project. These parameters are the same as those entered in Create new project. The project name cannot be changed.
	Data	The tool to set up the connections to the data sources and define the mappings to the tables in the target database.
	Transform	Provides tools for defining the data transformation rules.

(continues)

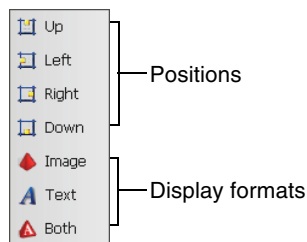
Table 2-2 Icons for the feature tabs (continued)

Icon	Label	Purpose
	Data Management	Allows the user to browse and explore the data in the target database, and create and remove links between data columns in tables.
	Admin	Supports administering the FastDB Engine users. An administrator can create a new engine user, change a password, and delete it. You must provide the following information: login name, password, password confirmation, and old password.

For security reasons, it is recommended that every user logs out before closing the browser.

Laying out the feature tabs

A tools menu provides customization options for the feature tabs in the user interface, as shown in Figure 2-7. These options support placing the tabs in different positions and displaying text, an icon, or both on each tab. To access these options, right-click any of the feature tabs.

**Figure 2-7** Options for tool tab placement and text

Running a load process

You start a load process by using a project in BIRT Analytics Loader. Before running a project, you must create the data source connections and define the loading transactions and the data transformation scripts. To make the loaded data available to the BIRT Analytics users, you must synchronize the loaded database with the active database and grant access privileges to database users.

After defining the table mappings and all transactions in a load process, you can run the project to load the data. You can run a single or multiple transaction processes. You can run a process with or without data transformation.

The load process loads the data in the FastDB database, the same database that users analyze. To avoid inaccurate analysis, consider loading data when users are not logged in to the system.

To prevent user access to databases while a load process runs, stop the web server running the BIRT Analytics application. Alternatively, schedule the loading process to run at a time when users are offline. For more information on how to schedule a load process, check “Scheduling load processes” in Chapter 6, “Administering BIRT Loader.”

Stopping the IIS service

BIRT Analytics installs on Microsoft Internet Information Services (IIS). To prevent end users from logging in to BIRT Analytics, stop the web server on which BIRT Analytics runs, using either a command-line editor or Windows Services.

Stopping IIS using Services

You use Services to stop, start, or restart a service.

How to stop IIS using Services

- 1 In Windows→Start, type:

```
services.msc
```

Services appears, as shown in Figure 2-8.

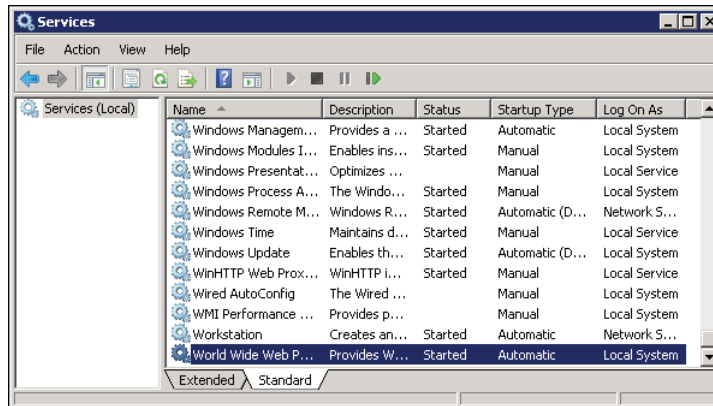


Figure 2-8 Stopping IIS using Services

- 2 In Services, select World Wide Web Publishing Service, then choose Stop from the context menu.

Stopping IIS using a command-line editor

To stop IIS from a command line, you use the Windows net command. You can also use these commands in batch processes if you need to automate the procedure.

How to stop IIS from a command line

- 1 Type cmd in Windows→Start to open a command prompt.
- 2 In the command-line editor, type the following command, then press Enter.

```
net stop w3svc
```

A response indicates that the service has stopped, as shown in Figure 2-9.

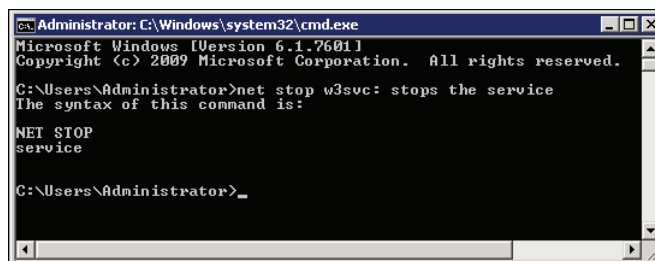


Figure 2-9 Stopping IIS from a command-line prompt

- 3 To restart the service after the load process finishes, type the following command at the next command prompt, then press Enter:

```
net start w3svc
```

Running a project

To run a loading project, you must choose Run, as shown in Figure 2-10.

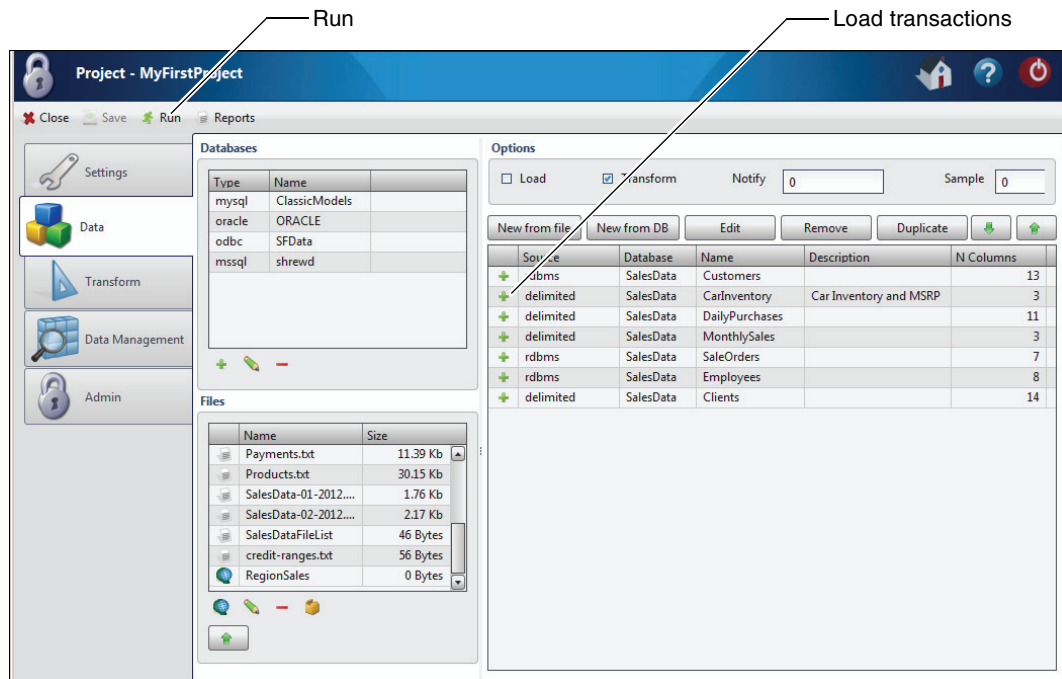


Figure 2-10 Running MyFirstProject

In the message that confirms whether you want to run the project, choose Yes. A list of database table names and the number of records loaded into each table appears in Running project, as shown in Figure 2-11.

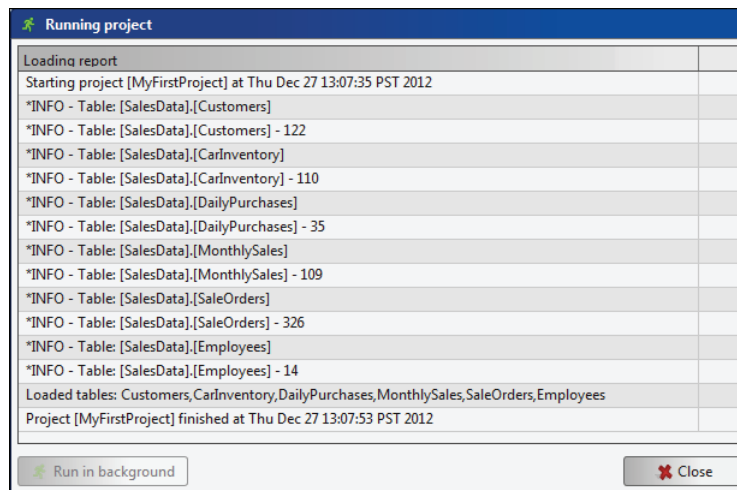


Figure 2-11 Running project log

The Running project log provides information about the tables into which the data is loaded, and the number of the records that are loaded. To limit the number of message lines in the log, set a number of records for which a log entry is created, in Options—Notify. For loading processes requiring a long time, choose Run in background, to minimize the Running project window. You can work on a different project until the running project finishes.

Using reports

To browse through the generated log files, select Reports from the control menu. The list of reports appear in Executions, as shown in Figure 2-12. Date displays a date and time when a project is executed. State displays the generation status. After selecting a date, the log content appears in Detail.

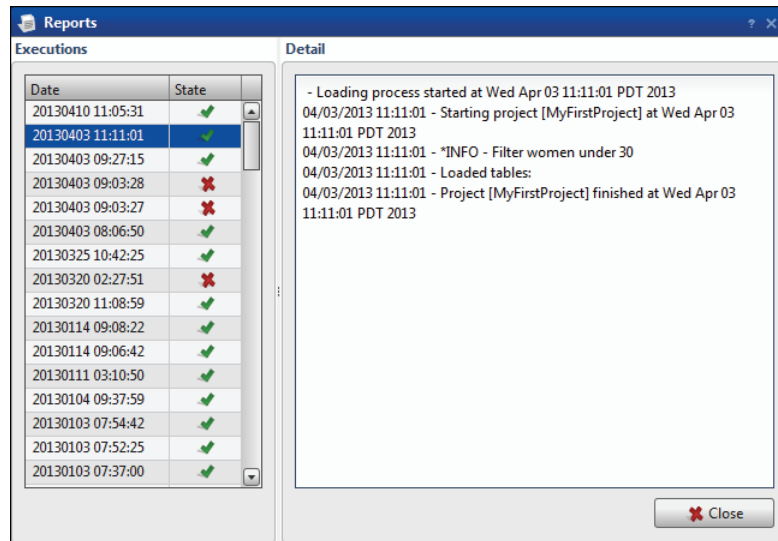


Figure 2-12 Browsing reports

Synchronizing data

Synchronizing data is the final step in the loading process. In BIRT Analytics, you can synchronize data two ways. You can use the BIRT Analytics Admin tool, or a command-line utility to automate the synchronization process.

Synchronizing data using BIRT Analytics Admin

The Synchronize option in BIRT Admin allows the administrator to update the application database to contain the most recent changes in the BIRT Analytics Engine repository. Execute this option after making a change in the analytical repository that alters the database structure, such as the addition, modification, or deletion of a column, table, or other data object. Synchronizing is also required when you change the data, after increasing, reducing, or modifying the records in a database.

After synchronizing the data you must set the privileges for the users or groups to provide access to the database. In the BIRT Analytics security options list, the ACL to objects option allows the administrator to create, modify, or delete a security group or access control list (ACL) and manage privileges over database objects.

How to synchronize data using BIRT Analytics Admin

To do the synchronization you must have access to the BIRT Analytics Admin application.

- 1 Using your web browser, connect to the following site:

`http://localhost/qadmin`

The BIRT Analytics login screen appears as shown in Figure 2-13.

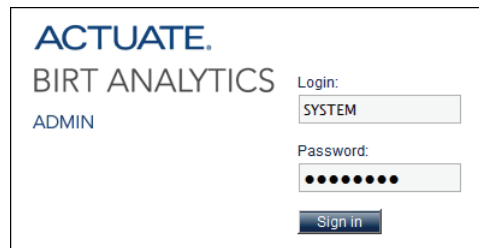


Figure 2-13 Logging in to BIRT Analytics Admin

- 2 In Login, type SYSTEM. In Password, type PASSWORD. Choose Sign in. The BIRT Analytics Admin page appears, as shown in Figure 2-14.

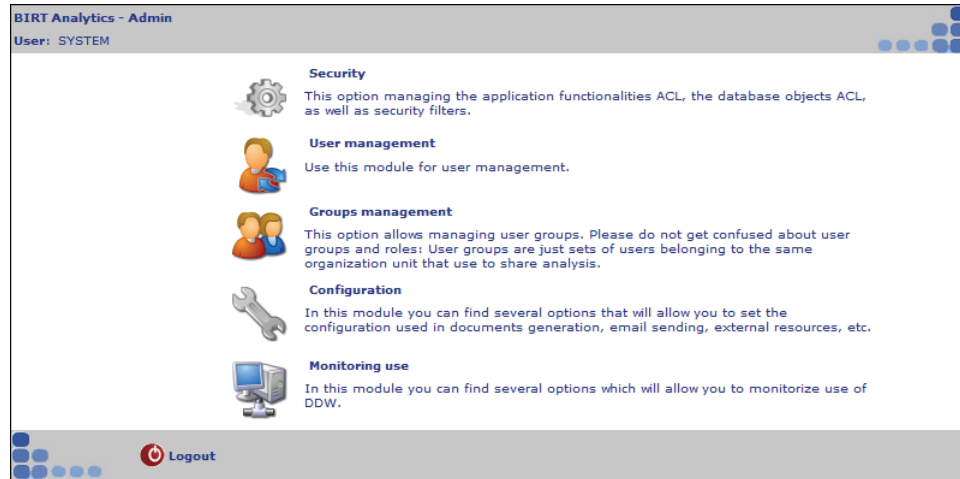


Figure 2-14 Accessing the security administration

- 3 In BIRT Analytics Admin, choose Security. The Security page appears, as shown in Figure 2-15.



Figure 2-15 Accessing the synchronize settings

- 4 In Security, choose Synchronize.
- 5 In Synchronize, choose Synchronize again, as shown in Figure 2-16.

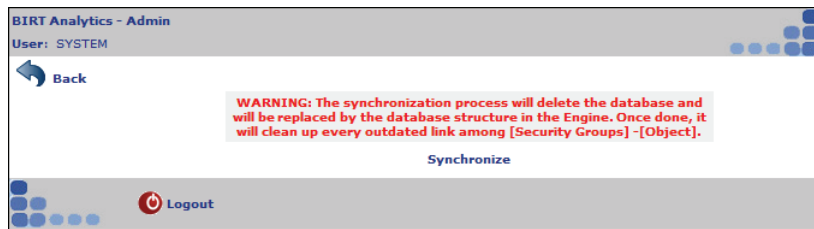


Figure 2-16 Updating the data structure from the engine

- 6 Choose Back when the synchronization has finished.
- 7 To make a synchronized database visible to a user security group, complete the following steps:
 - 1 In BIRT Analytics Admin, choose ACL to objects.
 - 2 Select a security group for which to assign access privileges. Then, choose Modify, as shown in Figure 2-17.

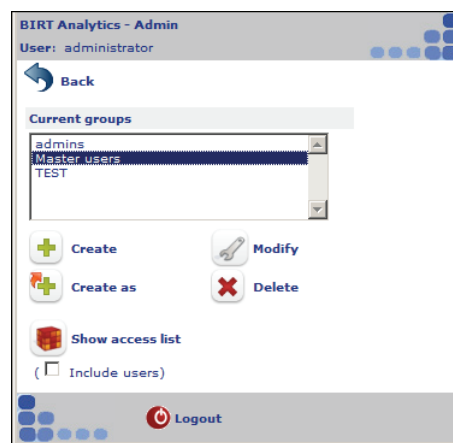


Figure 2-17 Selecting a user security group

A list of available databases appears, as shown in Figure 2-18.

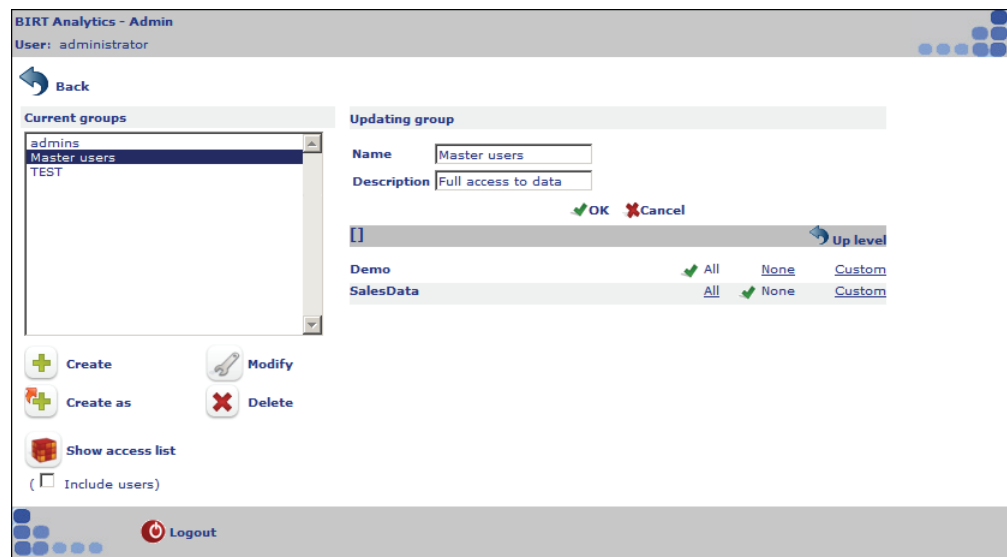


Figure 2-18 Granting a user group access privilege to a database

- 3 Select a database to which you will grant a user group access privileges. For example, select the SalesData database.

- 4 To grant a group full access to a database, choose All for that database. The modifications appear in Updating group. For example, Figure 2-19 shows a group called Master users, modified to have all privileges for, or full access to, the SalesData database.

Updating group		
Name	Master users	
Description	Full access to data	
<input type="button" value="OK"/> <input type="button" value="Cancel"/>		
<input type="button" value="Up level"/>		
Demo	<input checked="" type="checkbox"/> All	<input type="checkbox"/> None Custom
SalesData	<input checked="" type="checkbox"/> All	<input type="checkbox"/> None Custom

Figure 2-19 Updating group

- 5 Choose OK.
- 6 Choose Back.

How to verify functionality activation

To verify that all functionalities are activated for a user group, complete the following steps.

- 1 Choose a security role, as shown in Figure 2-20.
- 2 Select a user group, for example Administrators
- 3 Choose Modify.

BIRT Analytics - Admin
User: SYSTEM

[Back](#)

[Show access list](#)

Current roles

- Administrators
- Operators
- Read

[Logout](#)

Figure 2-20 Selecting the Administrators role

- 4 In Functionalities, de-select General. Then, select General, as shown in Figure 2-21.
- 5 Choose OK.

BIRT Analytics - Admin
User: SYSTEM

[Back](#)

[Show access list](#)

Current roles

- Administrators
- Operators
- Read

Updating role

Name: Administrators

Description: Administrators

Functionalities

☒ General

[Logout](#)

Figure 2-21 Selecting all functionalities

- 6 Choose Logout to exit BIRT Analytics Admin.

Synchronizing data using bashell

BIRT Analytics Loader provides a command-line tool named bashell that allows you to automate a synchronization process and run it in an unattended manner. Run bashell to synchronize an analytical database after loading data into FastDB. Table 2-3 lists the bashell parameters.

Table 2-3 bashell parameters

Parameter	Description
-site: <URL>	Specifies the web services URL of the BIRT Analytics application. For example, -site:http://localhost/qelectron.
-user: <login name>	Specifies the login name of the user who makes the request. It must be a valid administrator user in the BIRT Analytics application.
-pwd: <password>	Specifies the password of the user making the request.
-save	Causes bashell to save the current settings in an encrypted file named set.cfg that can be used for future requests.
-sync	Sends a request to the BIRT Analytics application to synchronize data.
-cacheoff	Removes cache files.
-help	Displays the help for this tool.

You can run bashell and authenticate with BIRT Analytics at the same time or you can create an encrypted authentication file and use it to automate the authentication. By default the authentication information is saved in a file named set.cfg.

How to run bashell

- 1 If you downloaded BIRT Analytics, open the 10-Tools folder and copy and paste the sub-folder named bashell and its contents to your desired folder.
- 2 Open a command prompt that is run as administrator in the bashell folder you created in the previous step.
- 3 Run the following command to synchronize the analytical database, where SYSTEM is an administrator user and PASSWORD is the password for that user:

```
bashell -sync -cacheoff -site:http://localhost/qelectron -user:SYSTEM -  
pwd:PASSWORD
```

How to create an authentication file

- 1 If you downloaded BIRT Analytics, open the 10-Tools folder and copy and paste the sub-folder named bashell and its contents to your desired folder.
- 2 Open a command prompt that is run as administrator in the bashell folder you created in the previous step.
- 3 Run the following command to create an encrypted authentication file, where SYSTEM is an administrator user and PASSWORD is the password for that user:

```
bashell -save -site:http://localhost/qelectron -user:SYSTEM -pwd:PASSWORD
```

- 4 Verify that the set.cfg file has been created in the bashell folder.

Once the set.cfg exists with a valid user and user password, you can use the tool without authenticating. For example you can run from command prompt or a batch file the following command to synchronize the analytical database and remove cache files:

```
bashell -sync -cacheoff
```

Managing FastDB Engine accounts

Dubnium server uses an internal account to communicate with the FastDB database and the rest of the internal modules. This account provides a security layer that protects sensitive data. The administrator who runs a loading project must create the secure account and login credentials for the account. BIRT Loader passes a project's user credentials to the FastDB Engine at run time. The engine authenticates the credentials against the existing secure database accounts. If an account does not exist or a password is not the same, a project fails to run and load data. This feature allows the system administrator to control user access to sensitive data by changing passwords or deleting FastDB Engine user accounts.

To manage FastDB Engine users, you use BIRT Loader Admin, as shown in Figure 2-22.

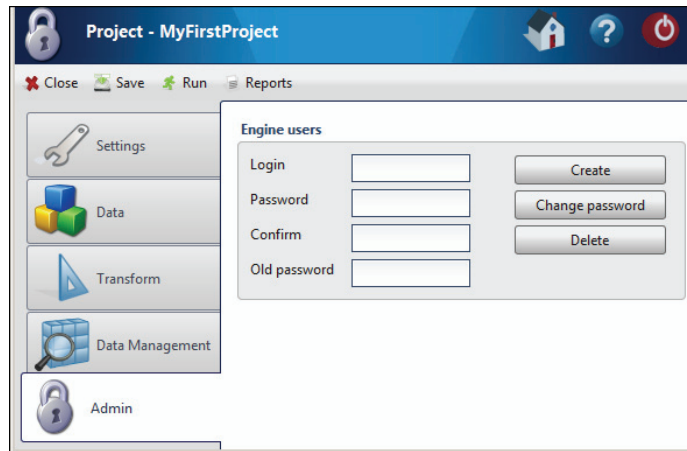


Figure 2-22 Managing FastDB Engine users

Use these Admin features to perform the following account creation tasks:

- **Create**
Create a new user account.
- **Change password**
Change a password of an existing user account.
- **Delete**
Delete an existing user account.

To create and manage each FastDB Engine user, perform the following tasks in all of these fields:

- **Login**
Define a user name.
- **Password**
Define a password.
- **Confirm**
Confirm the password. You use this property when you create a new user or change a password for an existing user.
- **Old password**
Define an old password. Required when you change a password.

3

Extracting data

This chapter contains:

- About extracting data
- Connecting to data
- Creating a load process

About extracting data

The load process is composed of two main steps: loading and transformation. The configuration of the load process is done in the Data module.

The Data page, as shown in Figure 3-1, is divided into three areas: Databases, Files, and Options. Databases and Files are places where you define the connections to the data sources. Options is the place where you define a load process. A load process consists of options that determine the way the load process runs, and transactions. A load transaction describes the data mappings between the input data and the tables in the target database. A load process may contain multiple transactions.

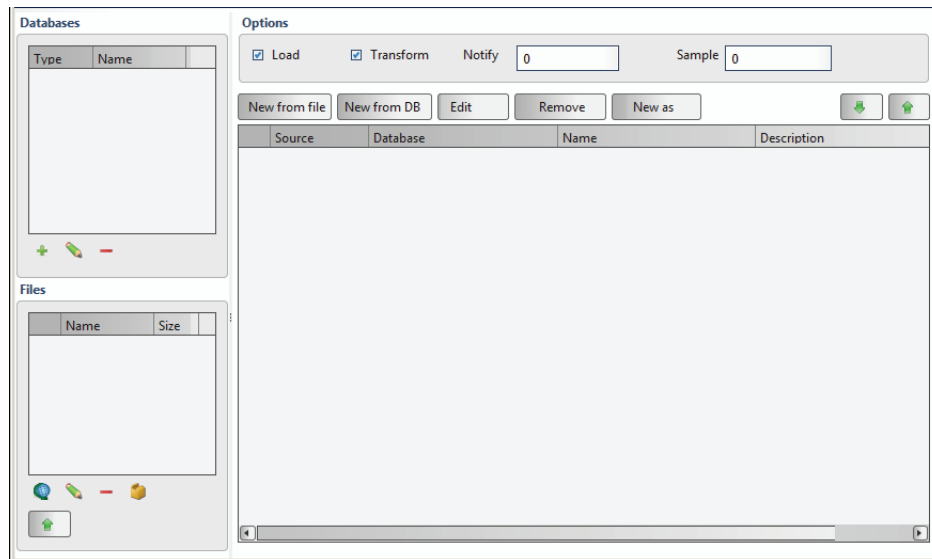


Figure 3-1 Extracting data

Connecting to data

BIRT Analytics Loader supports connections to relational databases and flat files. Data source connections are accessible across multiple projects associated with the same user. File connections are associated only with the project where they are defined. You cannot share file connections across multiple projects.

Connecting to databases

You can create and name as many data source connections as a load process requires. The data sources can be of different types. For example, a load process can use data from a database and data from a flat file repository. Each type of data source requires different connection information.

BIRT Analytics Loader connects to databases using native and ODBC drivers. BIRT Loader supports native drivers for the following database types:

- Microsoft SQL Server
- MySQL Database Server
- Oracle Database
- PostgreSQL

BIRT Analytics Loader also supports connecting to a database using an ODBC driver. To connect to a database having a native driver type not supported by BIRT Analytics Loader, you must provide an ODBC driver and set up an ODBC connection that uses the ODBC driver. In BIRT Analytics Loader you can define a new data source, edit an existing one, or remove a data source, as shown in Figure 3-2.

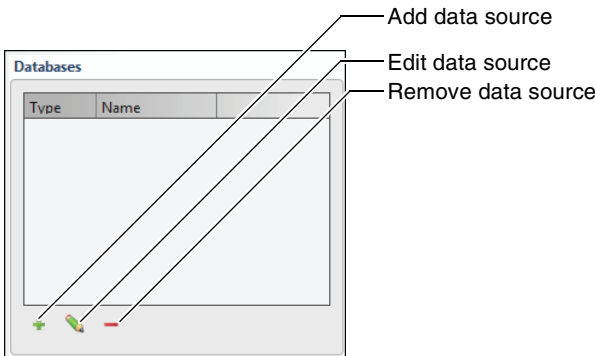


Figure 3-2 Connection tools for database data sources

Using native drivers

Before creating a connection using a native driver make sure the driver is installed on your machine. For MySQL server for example, check if the path to the driver is in the System Path. To see the System Path, use Control Panel→System→Advanced System Settings →Environment Variables→Path, as shown in Figure 3-3.

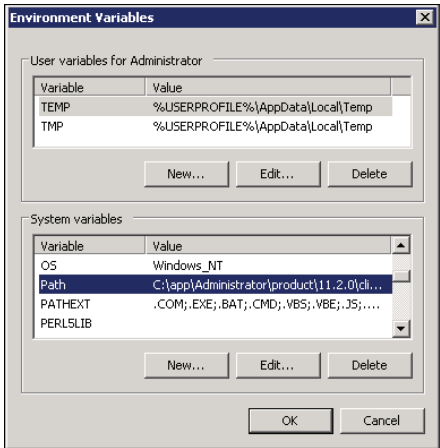


Figure 3-3 Editing Path variable

To create a data source connection you must provide the database connection information, as shown in Table 3-1. Contact your database administrator to obtain this information.

Table 3-1 Database connection properties

Property	Description
Name	The name of the database connection, which must be unique
Type	The type of connection, selected from the values in the drop-down list
Server	The database server, which can be a machine name or an IP address
Port	The connection port
User	A valid user name to connect to the database

(continues)

Table 3-1 Database connection properties (continued)

Property	Description
Password	A valid password to connect to the database
Database	The name of the database
Driver	The driver name, required for ODBC drivers only
Connection string	Custom connection properties, required for custom ODBC drivers only

How to connect to a database using a native driver



- 1 In Databases, choose the + sign. Datasource appears, as shown in Figure 3-4.

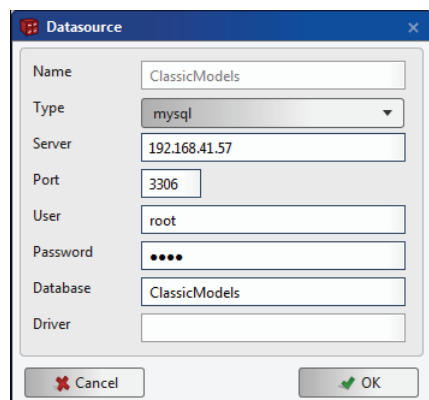


Figure 3-4 Using a native driver

- 2 In Datasource, define the following properties:
 - In Name, type a unique name. Using a name that describes the database content helps identify different databases you define as data sources.
 - In Type, select the source database type from the list of available types. In this example, mysql is the source database type.
 - In Server, type the name of the server that hosts the source database. In this example, an IP address identifies that server.
 - In Port, type the connection port. The mysql database installation in this example uses the default port 3306.
 - In User, type a valid user name defined for the source database.
 - In Password, type a valid password defined for the source database.
 - In Database, type the name of a database that contains source data.
 - In Driver, optionally type a native driver name, or type no characters. A connection that uses a native driver requires no name.
- 3 Choose OK to create the named data source connection. A new data source connection name and type appear in Databases, as shown in Figure 3-5.

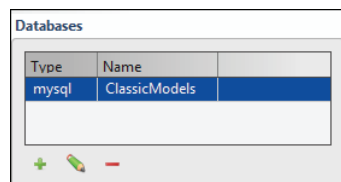


Figure 3-5 A data source connection that uses a native database driver

Using ODBC drivers

Creating a connection using an ODBC driver is very similar to the native driver connections. The difference is that you must provide a driver name as a connection property. As well as the predefined ODBC data source types, a custom ODBC data source is available. To use the custom ODBC data source, you must type the required connection parameters as a text value.

To determine the name of your ODBC driver, open Control Panel→ODBC Data Source Administrator. The driver name appears in the Drivers tab, as shown in Figure 3-6.

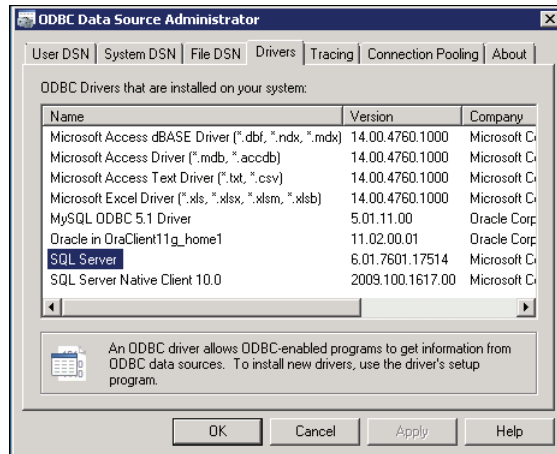


Figure 3-6 Finding an ODBC driver name

How to connect to a database using an ODBC driver



- 1 In Datasources, choose the + sign. Datasource appears.

In Datasource, define the following properties for a standard ODBC data source, as shown in Figure 3-7:

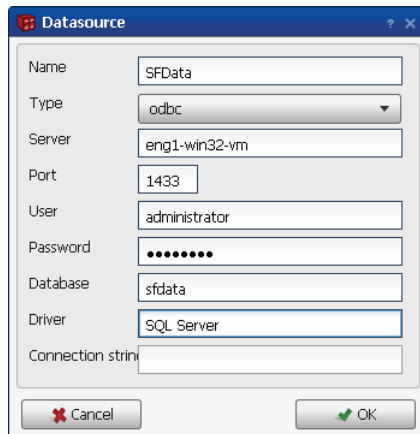


Figure 3-7 Using an ODBC driver

- In Name, type a unique name. Using a name that describes the database content helps identify different databases you define as data sources.
- In Type, select ODBC from the list of available connection types.
- In Server, type the name of the server that hosts the source database. In this example, a machine name, eng1-win32-vm, identifies that server.
- In Port, type the connection port. The mssql database installation in this example uses the default port, 1433.
- In User, type a valid user name defined for the source database.

- In Password, type a valid password defined for the source database.
- In Database, type the name of a database that contains source data.
- In Driver, type the driver name, as defined in the ODBC Data Source Administrator. In this example, the name of the ODBC Microsoft SQL Server driver is SQL Server.

Alternatively, for a custom ODBC data source, define the following properties, as shown in Figure 3-8:

Figure 3-8 Using a custom ODBC driver

- In Name, type a unique name. Using a name that describes the database content helps identify different databases you define as data sources.
 - In Type, select Custom ODBC from the list of available connection types.
 - In User, type a valid user name defined for the source database.
 - In Password, type a valid password defined for the source database.
 - In Database, type the name of a database that contains source data.
 - In Connection string, type the parameters required to connect to the custom data source.
- 2 Choose OK to create the named data source connection. A new data source connection name and type appear in Databases, as shown in Figure 3-9.

Type	Name
mysql	ClassicModels
odbc	SFDData

Figure 3-9 Database connections

Connecting to files

BIRT Analytics Loader can access data from a text file, or flat file, that conforms to a defined and consistent structure. These files are typically generated by business systems and applications that create logs. These files can also be spreadsheets saved as comma-separated values (CSV) files. Before using text file data in a load process, make sure the file uses a valid structure. The size of the text files is limited only by the limitations set by the operating system.

BIRT Analytics Loader supports the following options for defining flat file data sources:

- Uploading a file from a local system
- Uploading a file from a remote system
- Uploading multiple files using a file list

Using BIRT Analytics Loader - Files, you manage data uploads from data files stored on local and remote systems, using the toolset shown in Figure 3-10.

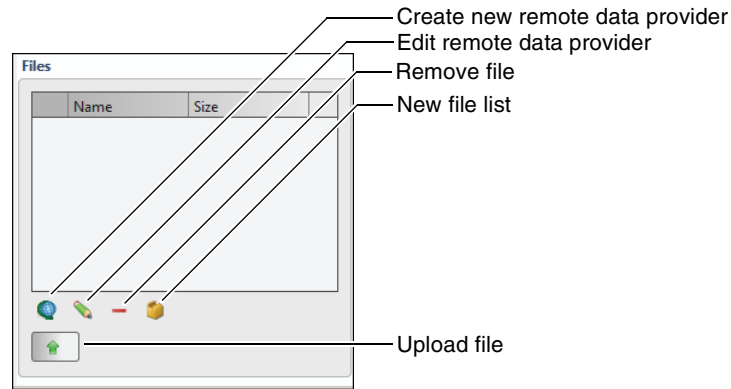


Figure 3-10 Connection tools for flat file data sources

To prepare to access your text files you must know the file name and location.

Uploading local files

After selecting the file, it will be uploaded into the data folder of the load project. Because of network reasons there is a configurable maximum file size available.

How to upload a file from a local file system



- 1 In Files, choose the up arrow. File Upload appears, as shown in Figure 3-11.

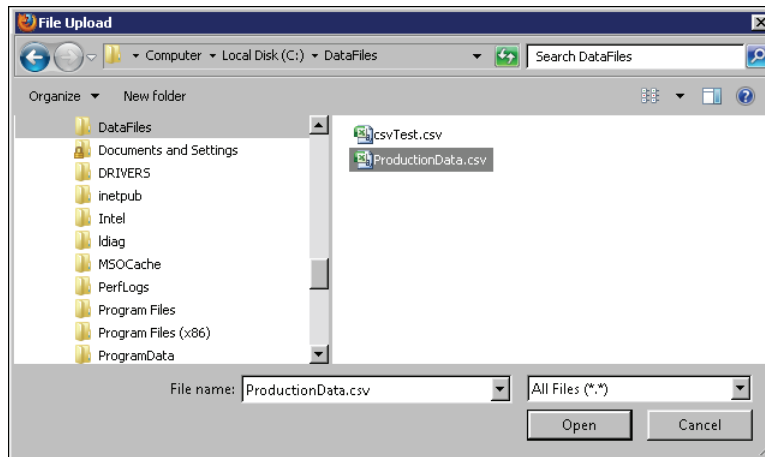


Figure 3-11 Uploading a local file

- 2 In File Upload, navigate to the folder where the file is located.
- 3 Select the file and choose Open. The file appears in Files, as shown in Figure 3-12. The local file is stored in the data folder of the project.

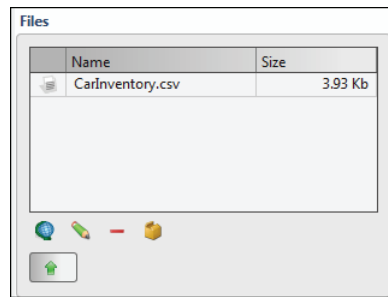


Figure 3-12 Creating a local file data source

Uploading remote files

The remote data provider option allows you to load files using a web address. You have to choose a name, the Uniform Resource Identifier (URI), and the HTTP method (GET, POST).

How to upload a file using URI



- 1 In Files, select the remote data provider icon. URI appears.
- 2 In URI, enter the following information, as shown in Figure 3-13:
 - In Name, type the name you want to assign to this data source.
 - In URI, type the URI, specifying the file location.
 - In Method, select the method for obtaining the file. Contact the file provider to identify the method.
 - In Parameters, enter parameters if needed.
 - Choose + to add parameter. In Name, type the parameter name. In Value, enter the parameter value, as shown in Figure 3-13.
 - To modify a parameter, select a column and change the entry.
 - Choose - to remove a parameter.

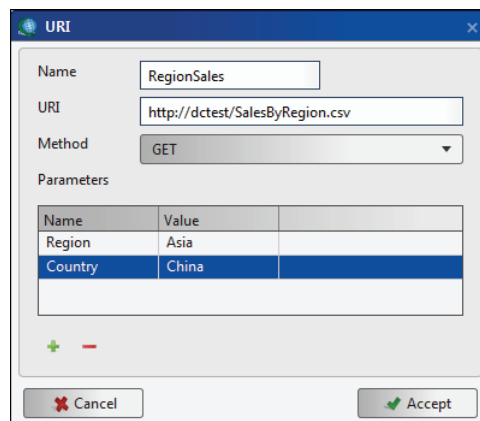


Figure 3-13 Providing URI properties

- 3 Choose OK. The new flat file data source appears in Files, as shown in Figure 3-14.

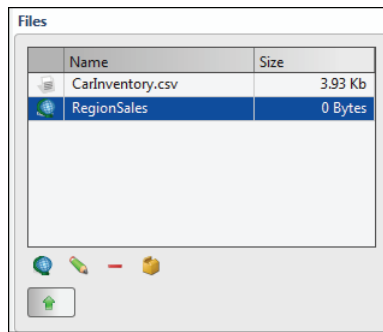


Figure 3-14 Creating a remote file data source

Uploading files using a file list

On many occasions, due to a large data volume and file size restrictions, it is convenient to break the data into multiple files. In such cases you must use a file list to create a file data source and upload the data into a single table.

How to upload files using a file list



- 1 In Files, upload all the files you want to include in a list, the same way you upload local files.
- 2 Press and hold Ctrl while using the mouse to select multiple file names, as shown in Figure 3-15.

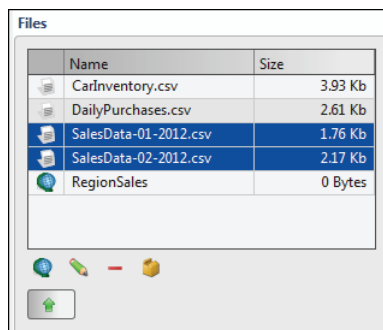


Figure 3-15 List file data source



- 3 In Files, choose New File list. File names that you selected in step 2 appear, as shown in Figure 3-16.

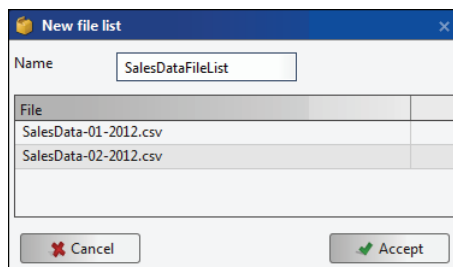


Figure 3-16 Selecting a list file

- 4 In Name, type a name for the new list file data source, then choose OK.
The list file data source appears in Files, as shown in Figure 3-17.

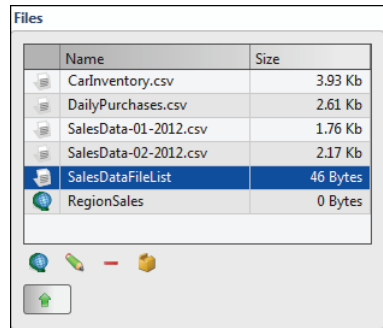


Figure 3-17 Naming a list file data source

Updating data source definitions



You can modify the data source definitions by choosing the pencil icon. The changes in the data source definitions require an update of the related table definitions. Saving the changes displays a message, as shown in Figure 3-18, asking you to confirm the update.

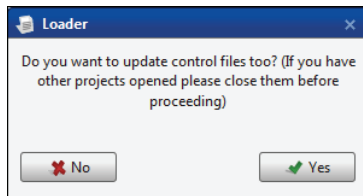


Figure 3-18 Updating data source message

Creating a load process

BIRT Analytics Loader supports creating data load transactions, setting load process options, and running a set of transactions as a process. To create a load process, first create the transactions that load data. Then, set options that run the load process.

About transactions

A transaction includes all settings and data mappings required to load data into a single table. A load process executes multiple transactions. You must save any change that you make to a load process before executing the process. To save any change to a process, choose Save on the menu bar.

BIRT Analytics Options, for defining a process and tools for creating transactions, looks like the example shown in Figure 3-19.

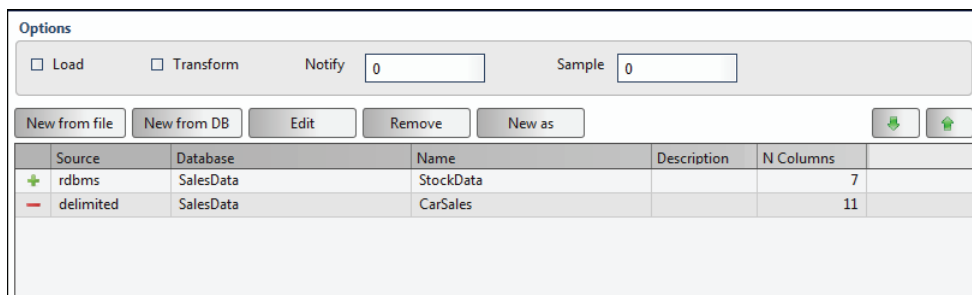
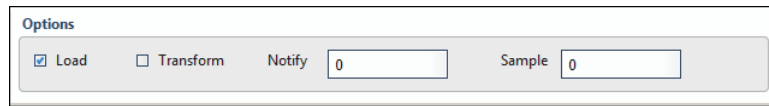


Figure 3-19 Defining data

Setting load process options

Use Options, shown in Figure 3-20, to select options that define a load process. Displayed settings that you save for each process execute when the process runs, as part of a data load project.



The 'Options' dialog box contains the following controls:

- ☒ Load
- ☐ Transform
- Notify:
- Sample:

Figure 3-20 Defining load process options

Select or specify values for each option to execute the following process tasks:

- **Load**
Select Load to execute all load transactions when a project runs.
- **Transform**
Select Transform to execute all transformation scripts when a project runs. Define transformation rules using the Transformation tab.
- **Notify**
Type a value that sets a trigger to write a log message each time the set number of records loads.
- **Sample**
Specify the number of records loaded for each table.

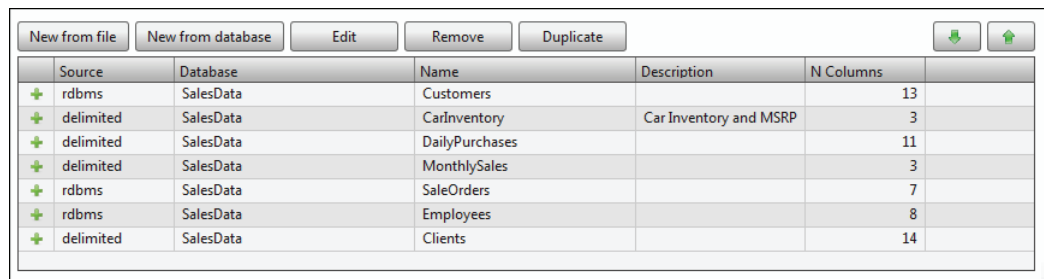
Creating a load transaction

Creating a load transaction involves defining the mappings between the source and target data objects. Mapping is the definition of the relationship and data flow between source and target objects.

Before building an actual data map for the loading process, you have to define a data model for target even if it is not an actual database. It will help you understand the entities to which it belongs. Then it will be easier to find the similar entity in the source and map it to the target element. This is called building a logical data map model and you have to do it before you actually start mapping the data.

After you have built a data map model for ETL (Execute, Transform, Load), you have to analyze the data content, complete business rules for ETL process, integrate data sources to a single source, and only then build an actual data map.

BIRT Analytics Loader supports creating multiple load transactions per project, as shown in Figure 3-21.



The interface includes buttons at the top: 'New from file', 'New from database', 'Edit', 'Remove', 'Duplicate', and two green arrow buttons. Below is a table with the following data:

	Source	Database	Name	Description	N Columns
+	rdbrms	SalesData	Customers		13
+	delimited	SalesData	CarInventory	Car Inventory and MSRP	3
+	delimited	SalesData	DailyPurchases		11
+	delimited	SalesData	MonthlySales		3
+	rdbrms	SalesData	SaleOrders		7
+	rdbrms	SalesData	Employees		8
+	delimited	SalesData	Clients		14

Figure 3-21 Creating multiple load transactions

You have the following choices when you create a load transaction:

- **New from file**
Define a transaction that creates a table mapped to a flat file connection.
- **New from database**
Define a transaction that creates a table mapped to a database.
- **Edit**
Edit a load transaction.
- **Remove**
Remove a load transaction.
- **Duplicate**
Create a new transaction using an existing one.

Mapping data from files

The mapping process requires defining the source and the target data. The software allows you to validate the data connection and create the table column definitions. You can automatically import the target table structure from the source file or manually create the column definitions. When the table structure does not map the source file data precisely, the loading process cannot identify the data to load for each table column and finishes unsuccessfully.

To map data from files you need to know how to do the following:

- Defining a target database
- Defining a source file format
- Defining table columns

The example in “How to map data from a CSV file” provides more information about the mapping process.

Defining a target database

To define the target database and table, and the way the data is loaded, you must define the following information, as shown in Figure 3-22:

The screenshot shows the 'Table declaration' dialog box. It has a title bar with a close button. The main area contains several groups of controls. On the left, there are dropdowns for 'Database', 'Type' (set to 'Delimited'), 'Encoding' (set to 'ISO-8859-1'), 'End Line' (set to 'CrLf'), 'File' (set to 'csvTestODA.csv'), and 'Delimiter'. To the right of these are 'Action' (set to 'Create'), 'Description' (text box), 'Skip' (set to '0'), 'Offset' (text box), 'Is list' (checkbox), 'View' (button), 'Get columns' (button), and 'Wellformed' (checkbox). At the top right are 'Overwrite' and 'Ignore' checkboxes. At the bottom are buttons for 'New', 'Edit', 'Remove', 'New as', 'Cancel', and 'Accept'. Below the buttons is a table with columns: Name, Type, Width, Decimals, Format, Index, Skip.

Figure 3-22 Creating a table mapping to a file

- **Database**
Type a name for the target database, or select a listed database name.

- **Name**
Type a unique name for a target data table.
- **Description**
Optionally, type characters that describe the target data table.
- **Action**
Select one of the following load operation types:
 - **Create**
Adds a new table and overwrites any content stored in an existing table, using the displayed table name.
 - **Append**
Adds records to any existing table that uses the displayed table name.
- **Overwrite**
Choose this option when the loading transaction overwrites the target table.
- **Ignore**
Choose this option when you must exclude the transaction from the load process.

Defining a source file format

Based on your knowledge of source file properties, select formatting properties that most closely match those of the source file. More accurately selecting source file properties minimizes manual definition of target table properties.

- **Type**
Select the file type. The files can be flat, or delimited. The flat files do not have delimiters and qualifiers.
- **Encoding**
Select an encoding type from the list. Table 3-2 lists and describes the file encoding types that BIRT Analytics supports.

Table 3-2 Supported encoding

Encoding	Description
ASCII	American Standard Code for Information Interchange
CP1252	Windows Latin-1
BIG5	Chinese character encoding method used in Taiwan, Hong Kong, and Macau for Traditional Chinese characters
BIG5-HKSCS	Standard supplementary character set used for information exchange in Chinese between the public and the Hong Kong SAR Government
CP1252	Windows Latin-1
GB18030	Chinese government standard describing the required language and character support necessary for software in China
GB2312	Official character set of the People's Republic of China, used for simplified Chinese characters
GBK	An extension of the GB2312 character set for simplified Chinese characters, used in the People's Republic of China
ISO-8859-1	ISO 8859-1, Latin Alphabet No. 1
ISO-8859-2	Latin Alphabet No. 2

(continues)

Table 3-2 Supported encoding (continued)

Encoding	Description
ISO-8859-3	Latin Alphabet No. 3
ISO-8859-4	Latin Alphabet No. 4
ISO-8859-5	Latin/Cyrillic Alphabet
ISO-8859-6	Latin/Arabic Alphabet
ISO-8859-7	Latin/Greek Alphabet
ISO-8859-8	Latin/Hebrew Alphabet
ISO-8859-9	Latin Alphabet No. 5
ISO-8859-13	Latin Alphabet No. 7
ISO-8859-15	Latin Alphabet No. 9
UTF8	8-bit UCS Transformation Format
UTF16	16-bit UCS Transformation Format, byte order identified by an optional byte-order mark
UTF16LE	16-bit Unicode Transformation Format, little-endian byte order
UTF16BE	16-bit Unicode Transformation Format, big-endian byte order
WINDOWS-1252	Windows Latin-1

- **End Line**
Select an end-of-line character from the list. Table 3-3 lists and describes the end-of-line characters and results supported by BIRT Analytics Loader.

Table 3-3 End-of-line characters and results

End-of-line character	Result description
CRLF	Carriage return and line feed
CR	Carriage return only
LF	Line feed only

- **File**
Select a source file name from the list.
- **Delimiter**
Select the character used to delimit or separate values in a delimited file only. The following delimiter characters are supported by BIRT Loader:
, ->| | : ; @ # " + - = '
- **Header**
Select Header option if the source file includes information in a header. A file header contains multiple column names, each separated by a delimiter character.
- **Skip**
In Skip, type a number that sets how many blank rows to insert between header information and record information in a file. If you select Header option, 1 appears in Skip by default.
- **Offset**
Define the initial position of each column when the file format is flat file.

- **Qualifier**
Select a qualifier character. BIRT Loader supports the following qualifier characters:
" ' ~
Choose None, if the file format does not use a qualifier.
- **Wellformed**
Select the Wellformed option to enforce verification that each record contains all columns.

Defining table columns

To automate defining the structure of a target data table, extract the data from a source data file, using Get columns. During extraction, Get columns adds column type and index information to each data element. You must review and edit, as necessary, the type and index information for every column. Consider removing the index from every column not strictly requiring an index. Removing unnecessary indexes speeds data load and analysis processes, but may limit performing aggregation and link operations. These operations require indexed columns.

- **View**
Choose View to see the data in the file and check the connectivity to the file.
- **Get columns**
Choose Get columns to get the column names and type automatically from the source file.

To create a target data table manually, define each column using the New, Edit, Remove, and Duplicate options. While you edit the structure of a target data table, use Get columns to retrieve existing column information, then use Edit to make modifications. To create a new column, choose New, then provide information using Column Declaration, as shown in Figure 3-23.

The screenshot shows a 'Column declaration' dialog box with the following fields and options:

- Name:** A text input field.
- Description:** A text input field.
- Type:** A dropdown menu with 'String' selected.
- Size:** A text input field.
- Format:** A dropdown menu with 'None' selected.
- Decimals:** A text input field.
- Dec.separator:** A dropdown menu with 'Point' selected.
- Indexed:** A checked checkbox.
- Skip:** An unchecked checkbox.
- Buttons:** 'Cancel' and 'OK' buttons at the bottom.

Figure 3-23 Creating a new column

- **New**
 - Creates a new column in the target table
 - **Name**
Enter the name of the column.
 - **Description**
Enter a description of the column.

- **Type**
Enter the data type. The supported data types are described in Table 3-4.

Table 3-4 Data types

Type	Description
String	A sequence of characters, up to 1024.
Integer	Integer (whole number) data from -2^{31} (-2,147,483,648) through $2^{31} - 1$ (2,147,483,647).
Longint	Integer (whole number) data from -2^{63} (-9,223,372,036,854,775,808) through $2^{63}-1$ (9,223,372,036,854,775,807).
Real	Floating precision number data with the following valid values: $-3.40E + 38$ through $-1.18E - 38$, 0 and $1.18E - 38$ through $3.40E + 38$.
Date	The default format is mm_dd_yyyy, where the date separator can be slash (/), comma (,), period (.), or no separator.
Datetime	Date and time data from January 1, 1400, through December 31, 9999, with an accuracy of three-hundredths of a second, or 3.33 milliseconds. The default format is yyyy_mm_dd_hh_MM_ss.
Time	The default format is hh_MM_ss.
Unicode	A sequence of Unicode characters, up to 1024.

- **Size**
The size of the column in characters, or digits.
- **Format**
The format of the columns of type Date, Datetime, and Time.
- **Decimals**
The number of digits to the right of the decimal point, or comma. Applies for data of type Real.
- **Decimal separator**
You can choose Point or Comma to visualize a decimal separator. Applies for data of type Real.
- **Indexed**
Indicates if the column is an index.
- **Skip**
Select Skip to exclude a new column from the load process.
- **Edit**
Opens the Table Declaration editor for a single column.
- **Remove**
Removes a column from the target table.
- **Duplicate**
Creates a new column by copying the properties of an existing column.

How to map data from a CSV file

The following procedure assumes that you have created a data file connection. The procedure uses as an example, a CSV file having the following structure:

```

PRODUCTNAME,QUANTITYINSTOCK,MSRP
1969 Harley Davidson Ultimate Chopper,7933,95.7
1952 Alpine Renault 1300,7305,214.3

```

- 1 In Options, choose New from file.
- 2 In Table Declaration, set properties for the target database, as shown in Figure 3-24.
 - 1 In Database, type a database name. Alternatively, select an existing database name from the list of target databases.
 - 2 In Name, type the name a target table to which you will map data. For example, CarInventory.
 - 3 In Description, type characters that describe information the table will contain, such as Car Inventory and MSRP.
 - 4 In Action, select Create.
 - 5 Select Overwrite.

Figure 3-24 Defining file structure

- 3 Select options that match properties of the source file, as shown Figure 3-24.
 - 1 In Type, select Delimited.
 - 2 In Encoding, select ASCII.
 - 3 Select Header. In this example, 1 appears in Skip. Edit the value in Skip to match the number of header rows in the source data file.
 - 4 In End Line, select CrLf.
 - 5 In File, select a source file name from the list.
 - 6 In Delimiter, select comma (,).
 - 7 In Qualifier, select ".
 - 8 Select Wellformed.
- 4 Choose View to validate the data connection. The file content appears, as shown in Figure 3-25. Choose OK.

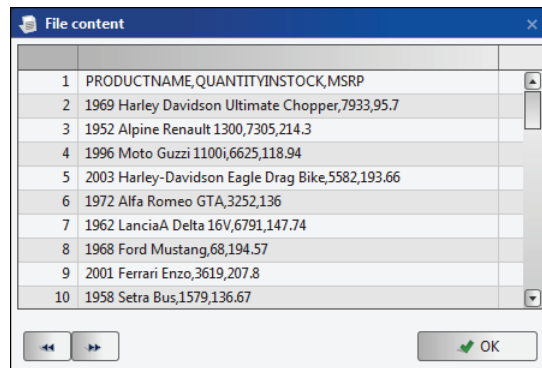


Figure 3-25 Viewing file content

- 5 Choose Get columns to import the file metadata.

During the metadata import process, Get columns reads column name information from the header row in the source file, determines a type for each column, and indexes each column.

Definitions for each new column created in the target table appear listed in Table Declaration, as shown in Figure 3-26.

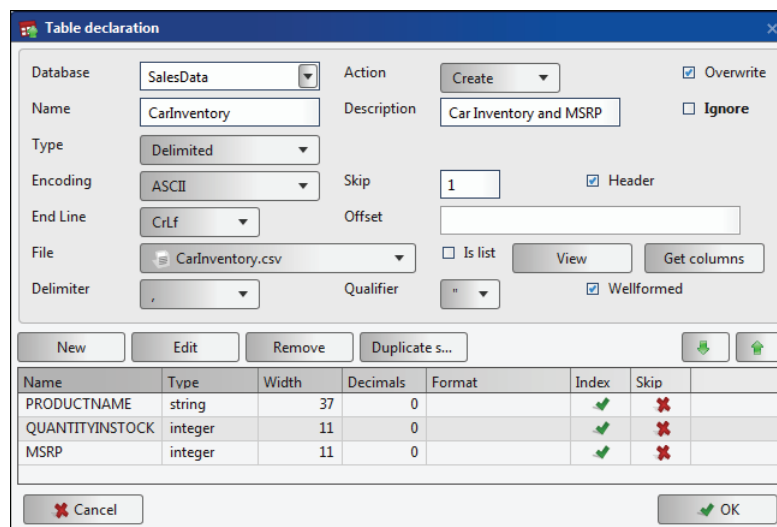


Figure 3-26 Getting column definitions

- 6 To modify properties for a column, select a listed column definition and choose Edit.
 - 1 In Column Declaration, select or modify any available properties.
 - 2 To remove index information from a column, deselect Indexed.
 - 3 To exclude a column from loading, select Skip.
 - 4 Choose OK.

For example, change the type of the MSRP column from Integer to Real, and set the column to not indexed, as shown in Figure 3-27.

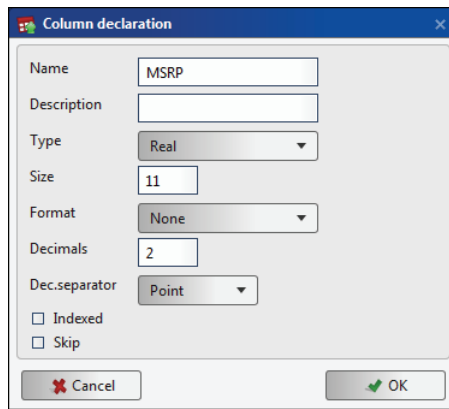


Figure 3-27 Editing column definition properties

- 7 To close a selected database table, in Table Declaration, choose OK.
- 8 In Project, choose Save. Choose Yes to confirm saving the script for the current project, as shown in Figure 3-28.

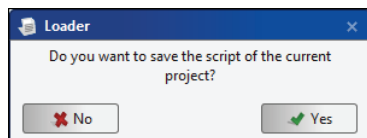


Figure 3-28 Saving a project

Mapping data from databases

Mapping data from a database requires that you define many of the same properties as those that you define for mapping data from files. To map data from a database, first define the target database, provide a target table name, and choose a database data source. If necessary, to review the procedure for setting these properties, see “Defining a target database,” earlier in this chapter. Next, define the data set to retrieve from the source database using Table Declaration, as described in Defining a data set. Selecting a database in Datasource provides the option to define a query that retrieves a data set.

The example in “How to map data from an Oracle data source” provides more information about this process.

Defining a data set

To define a data set you must provide a query and set options for the query.



■ Query

Defines a SQL query from a relational database data source. You can type a query manually, and use Get columns to retrieve the columns from the data set.

Alternatively, you can use View database structure to select tables and columns. Use Datasource to select tables and their columns, and specify the column order.

■ Options

Enter the options required by the connection drivers. For example, when loading data from an Oracle database, it is recommended to include a parameter, such as `PreFetchRows=10000`.

The process of defining the target table structure is the same as the one discussed in “Defining table columns,” earlier in this chapter. While you edit the structure of a target data table, use Get columns to retrieve existing column information, then use Edit to make modifications. To create a new column, choose New, then provide information using Table Declaration, as shown in Figure 3-22.

How to map data from an Oracle data source

This procedure assumes you have already created an Oracle database connection.

1 Define the target database and table.

- 1 In Options, choose New from DB. Table Declaration appears, as shown in Figure 3-29.

The 'Table declaration' dialog box is shown. It contains the following fields and controls:

- Database:** A dropdown menu.
- Name:** A text input field.
- Datasource:** A dropdown menu showing 'ClassicModels'.
- Query:** A large text area.
- Options:** A text input field.
- Action:** A dropdown menu showing 'Create'.
- Overwrite:** A checkbox.
- Ignore:** A checkbox.
- Buttons:** 'View', 'Get columns', 'New', 'Edit', 'Remove', 'Duplicate', 'Cancel', and 'OK'.

Figure 3-29 Declaring a table from a database

- 2 In Table Declaration, choose the target database from the drop-down list, as shown in Figure 3-30.

The 'Table declaration' dialog box is shown with the following updates:

- Database:** The dropdown menu is now set to 'SalesData'.
- Name:** The text input field now contains 'Scott'.
- Datasource:** The dropdown menu is now set to 'ORACLE'.

Figure 3-30 Mapping data from an Oracle database

- 3 In Name, specify the name of the target table.
 - 4 In Description, enter the table description.
 - 5 In Action, choose Create.
 - 6 Select Overwrite.
- ### 2 Define the data set.
- 1 In Datasource, choose the database data source.
 - 2 In Query, choose View database structure to build the query. Datasource appears.
 - In Datasource, choose a table from the table drop-down list. The column names appear, as shown in Figure 3-31.
 - Use + and - to add or remove columns, and the up and down arrows if you need to change the column order.
 - In Datasource, choose OK. The column definitions appear, as shown in Figure 3-31.

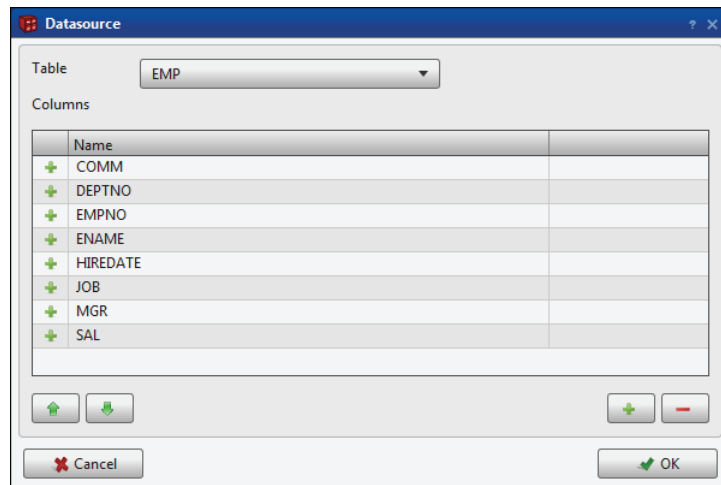


Figure 3-31 Using the query builder

- 3 In Options, specify any options you wish to pass to the data driver.
- 3 Choose View to validate the data connection and view the data. Query results appears, as shown in Figure 3-32.

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	1980-12-17T00:00:00	800		20
7499	ALLEN	SALESMAN	7698	1981-02-20T00:00:00	1600	300	30
7521	WARD	SALESMAN	7698	1981-02-22T00:00:00	1250	500	30
7566	JONES	MANAGER	7839	1981-04-02T00:00:00	2975		20
7654	MARTIN	SALESMAN	7698	1981-09-28T00:00:00	1250	1400	30
7698	BLAKE	MANAGER	7839	1981-05-01T00:00:00	2850		30
7782	CLARK	MANAGER	7839	1981-06-09T00:00:00	2450		10
7844	TURNER	SALESMAN	7698	1981-09-08T00:00:00	1500	0	30
7876	ADAMS	CLERK	7788	1987-05-23T00:00:00	1100		20
7934	MILLER	CLERK	7782	1982-01-23T00:00:00	1300		10

Figure 3-32 Viewing data from a database

- 4 Choose Get columns to import the columns from the data set. The column definitions appear as shown in Figure 3-33. The types are determined automatically, and all columns are indexed by default. Get columns overwrites the existing data set, and discards your existing changes.

Name	Type	Width	Decimals	Format	Index	Skip
EMPNO	integer	22	0		✓	✗
ENAME	unicode	10	0		✓	✗
JOB	unicode	9	0		✓	✗
MGR	integer	22	0		✓	✗
HIREDATE	datetime	7	0	yyyy-mm-dd_hh_MM_ss	✓	✗
SAL	real	22	2		✓	✗
COMM	real	22	2		✓	✗
DEPTNO	integer	22	0		✓	✗

Figure 3-33 Getting the column definitions from the data set

- 5 Edit a column.

The example in Figure 3-34 shows changing the format of the HIREDATE column, and setting the column as not indexed.

The screenshot shows a 'Column declaration' dialog box with the following fields and settings:

- Name: HIREDATE
- Description: (empty)
- Type: Datetime (selected in the dropdown)
- Size: 7
- Format: mmddyyyy_hhMMss (selected in the dropdown)
- Decimals: 0
- Dec.separator: Point (selected in the dropdown)
- ☐ Indexed
- ☐ Skip

At the bottom, there are 'Cancel' and 'OK' buttons.

Figure 3-34 Editing a Datetime column

- 1 Select the HIREDATE column and choose Edit.
- 2 In Type, select Date format.
- 3 In Format, select mm_dd_yyyy from the drop-down list.
- 4 Deselect Indexed. Choose OK.
- 6 In Project, choose Save. Choose Yes to save the project.

4

Managing data

This chapter contains:

- About managing data
- Understanding Explorer
- Understanding dynamic and permanent columns
- Understanding Links

About managing data

BIRT Loader Data Management, shown in Figure 4-1, provides tools for exploring data and creating and maintaining links between tables. BIRT Analytics Loader connects directly to the database engine. You explore the structure of all databases currently stored in FastDB, without synchronizing, using BIRT Analytics Loader Explorer. Explorer displays data objects, such as databases, tables, columns and their properties, and the discrete values of each column. The Links tool creates and maintains links between tables.

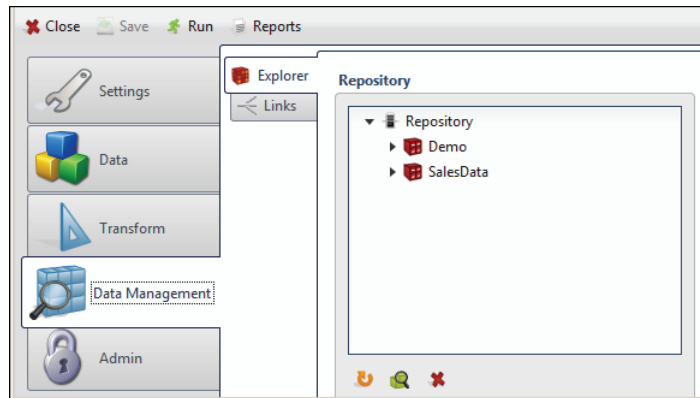


Figure 4-1 Managing data in FastDB

Understanding Explorer

Use Explorer to select and browse database structures stored in FastDB. Explorer supports browsing of multiple database objects, such as databases, tables, and columns. Explorer also supports refreshing the list of database names, viewing properties of a selected table and viewing properties of a selected column, using the features shown in Figure 4-2.

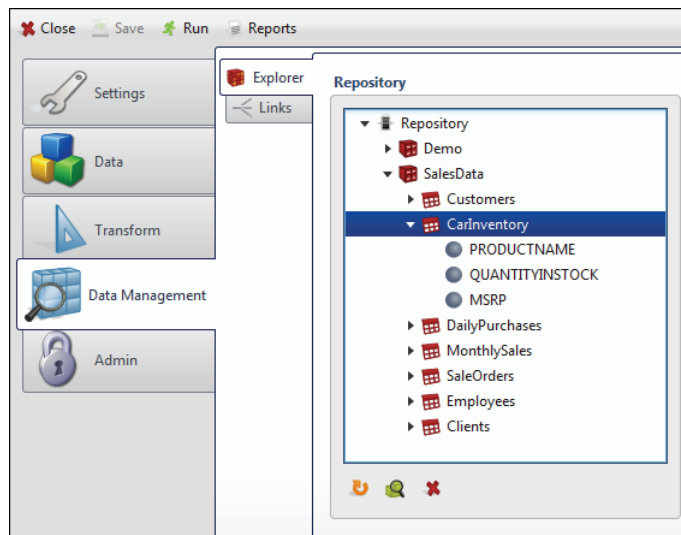


Figure 4-2 Exploring data structures in FastDB

How to view data structures stored in FastDB

- 1 In BIRT Analytics Loader, choose Data Management, and then Explorer.
The repository tree lists all database names currently stored in FastDB.

- 2 To view a list of tables in a database, in Repository, double-click a database name.
The database link expands and lists all table names defined in the selected database.
- 3 To view a list of database columns, in Repository, double-click a table name.
The table link expands and lists all column names defined in the selected table.

Refreshing the repository content



In Repository, choose Refresh to ensure that you see all databases, tables, and their columns currently stored in FastDB.

Exploring table properties



To view table properties, in Repository, select a table name, then choose View details. For a selected table, Table displays a value for each of the following table properties:

- ID
Internal table identification number
- Name
Table name
- Full name
Full table name in format [Database].[Table]
- Description
Table description
- Rows
Number of rows in a table
- Columns
Number of columns in a table

For example, Figure 4-3 shows all properties defined for the CarInventory table in the SalesData database.

The screenshot shows a dialog box titled 'Table' with a close button (X) in the top right corner. It contains several labeled text input fields:

ID	90
Name	CarInventory
Full name	[SalesData].[CarInventory]
Description	Car Inventory and MSRP
Rows	110
Columns	4

At the bottom of the dialog is an 'OK' button with a green checkmark icon.

Figure 4-3 Exploring table properties

Exploring column properties

To view column properties, in Repository, select a column name, then choose View details. For a selected column, Column displays Attributes and Discretes.

Attributes displays a value for each of the following column properties:

- ID
Internal code that identifies the column

- **Name**
A column name
- **Full name**
Full column name in the following format:
[Database] . [Table] . [Column]
- **Type**
Type of data (text, integer, and so on)
- **Discretes**
Number of column's discrete values
- **Rows**
Number of rows in the database
- **Nulls**
Number of column's null values
- **Size**
Column size
- **Indexed**
Indicates if column is indexed
- **Derived**
Indicates if the column is dynamic or permanent

For example, Figure 4-4 shows all properties defined for the Class Code column in the Household table.

The screenshot shows a 'Column' dialog box with two tabs: 'Attributes' (selected) and 'Discretes'. The 'Attributes' tab contains the following properties and values:

ID	18
Name	Class Code
Full name	[Demo].[Household].[Class Code]
Type	String
Discretes	12
Rows	226932
Nulls	0
Size	4
Indexed	<input checked="" type="checkbox"/>
Derived	Not derived

An 'OK' button with a green checkmark is located at the bottom right of the dialog.

Figure 4-4 Exploring properties of a permanent column

Discretes displays a value for each of the following column properties:

- **Value**
Discrete value of the column
- **Count**
Number of records for which the column has this value

- Percentage

Percentile of the count number to the total number of records

For example, Figure 4-5 shows all discrete values defined for the Class Code column in the Household table.

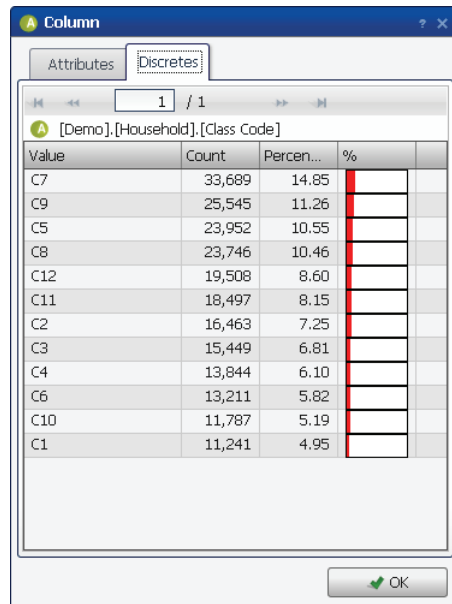


Figure 4-5 Exploring discrete values of a permanent column

Dropping database objects



To drop databases, tables, or data columns, select the data object in the repository tree, and then choose Drop. Drop removes the data object from the FastDB database.

Understanding dynamic and permanent columns

BIRT Analytics uses the term dynamic column to describe a column that contains only a formula, but no data values. A formula consists of one or more expressions and operators. Using BIRT Analytics Loader, you can edit the formula that defines a dynamic column. To edit such a formula, you change any of the expressions or operators in the formula. You create a dynamic column using instructions known as transformation instructions. The dynamic columns appear as editable columns in BIRT Analytics.

For a dynamic column, the column property Derived displays the formula name of the translation instruction that you use to create the column. For example, Figure 4-6 shows the column properties for a dynamic column, [TotalInStock]. It was created using the translation instruction, Expression. The Derived property displays Expression.

BIRT Analytics uses the term permanent column to describe a column that contains data values. You typically load the data values in a permanent column from an external database. You can transform a dynamic column to a permanent column using the transformation instruction, MAKEPERMANENT. For more information about transformations, see Chapter 5, "Transforming data."

ID	577
Name	TotalInStock
Full name	[SalesData].[CarInventory]
Type	Integer
Discretes	110
Rows	110
Nulls	0
Size	11
Indexed	<input checked="" type="checkbox"/>
Derived	Expression

Figure 4-6 Exploring properties for a derived column

Understanding Links

You link tables so that records from one table match related records from another. When you link, you are using a column that is common to both tables. BIRT Analytics uses the link to match up records from one table with those from the other. For example, if you use an Orders table and a Customers table, you link the tables so that each order from the Orders table can be matched up with the customer from the Customers table that made the order. In this example, the link assures that the data in each row of the report refers to the same order.

Links, shown in Figure 4-7, displays final and transient links. Final are the links you create using the Links option or the transformation instruction, LINK. For more information on LINK, see “LINK” in Chapter 5, “Transforming data.” Transient links are automatically generated by the software for performance reasons. For example, the transient link, shown in Figure 4-7, gets created if you create a crosstab with [Customer].[Occupation] as a dimension and Count(Order Detail) as a measure.

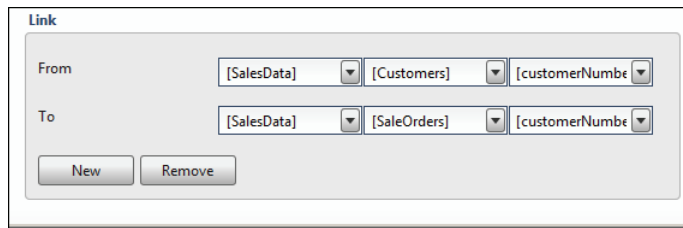
To clear the transient links, choose Remove all transient links.

Type	From	To
Final	[Demo].[Household].[Household ID]	[Demo].[Customer].[Household ID]
Final	[Demo].[Customer].[Cust ID]	[Demo].[Order].[Cust ID]
Final	[Demo].[Order].[Order No]	[Demo].[Order Detail].[Order No]
Transient	[Demo].[Customer]	[Demo].[Order Detail]

Figure 4-7 Managing links

How to create a new link

- 1 In Link, in From, select the database, table, and column from which the link starts.
- 2 In Link, in To, choose the database, table, and column to which the link goes, as shown in Figure 4-8.



The screenshot shows a dialog box titled "Link". It contains two rows of dropdown menus. The first row is labeled "From" and has three dropdown menus with the following selected values: "[SalesData]", "[Customers]", and "[customerNumbe]". The second row is labeled "To" and has three dropdown menus with the following selected values: "[SalesData]", "[SaleOrders]", and "[customerNumbe]". Below the dropdown menus are two buttons: "New" and "Remove".

Figure 4-8 Creating links

- 3 Choose New to create the link. Choose Yes in the message asking you to confirm. The new link appears in the list of links.

How to remove a link

- 1 In Link, in From, select the database, table, and column from which the link starts.
- 2 In Link, in To, choose the database, table, and column to which the link goes.
- 3 Choose Remove to drop the link. Choose Yes in the message asking you to confirm. The link disappears from the list of links.

Transforming data

This chapter contains:

- About transforming data
- Creating new columns
- Managing data objects
- Miscellaneous instructions

About transforming data

Transformation is the process of manipulating data. Any manipulation beyond copying is a transformation. Examples include cleansing, aggregating, and integrating data from multiple sources.

Transforming the data is executed after the load process is finished. The transformation script consists of one or more script files. The list of the script files is shown in Files, as shown in Figure 5-1. The Main file is the first file executed after data load. To access this file, you select Main file.

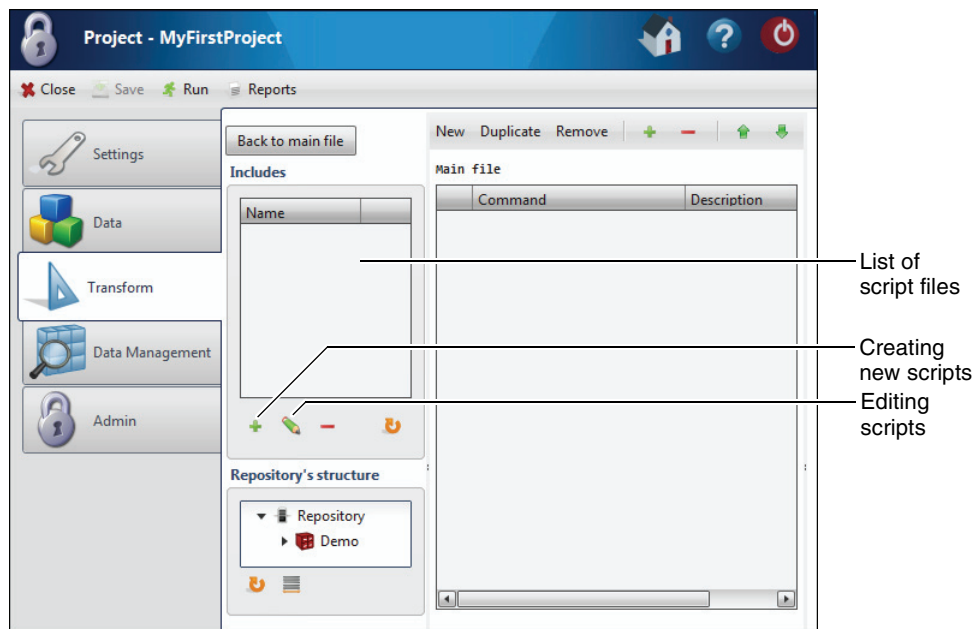


Figure 5-1 Transforming data

A new form for creating a new transaction opens on the right side of the Main file area. You can resize each of the three areas on the Transform screen by positioning your pointer on the three-dot line separating the areas, and dragging the pointer in the desired direction.

Understanding Includes

You use Includes to create script files. To execute these script files, you must include them in the Main file script.

- Opens a new Include file
Choosing Open requests a new file name, as shown in Figure 5-2.

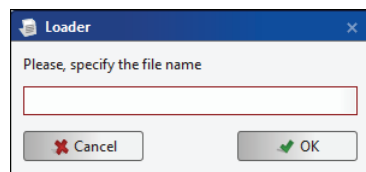


Figure 5-2 Creating a new script

- Edits a script file
Selecting Edit opens the file in the editing script area as shown in Figure 5-3.

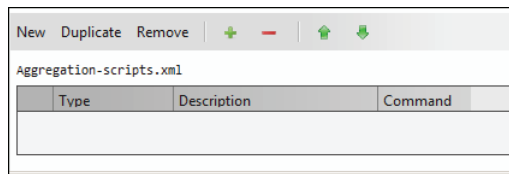




Figure 5-3 Creating instructions for a new script

You can copy, paste, and delete instructions in Include files using the shortcut keys, shown in Table 5-1.

Table 5-1 Editing shortcut keys

Functionality	All browsers shortcut keys	Firefox-only shortcut keys
Copy	Ctrl+Insert	Ctrl+C
Paste	Shift+Insert	Ctrl+V
Cut	Shift+Delete	Ctrl+X

-  ■ Removes a file
-  ■ Refreshes the file list in Includes

Understanding scripts

Choosing Main file opens the main script in the editing area. Each script file contains transformation instructions. To create these instructions you use the following options:

- **New**
Create a new instruction. When you choose New to create a new transformation instruction, the list of instructions appears, as shown in Figure 5-4. You select an instruction from the list, and provide the appropriate parameters for the instruction.







Figure 5-4 Transformation instructions

Depending on their purpose, the transformation instructions are grouped in three categories:

- Instructions for creating new columns
- Instructions for managing data objects
- Miscellaneous instructions

For more information on the transformation instructions, see *Using BIRT Analytics*.

- Duplicate
Create a new instruction based on an existing one.
- Remove
Remove an instruction.
-  ■ Enable
Enable an instruction for execution.
-  ■ Disable
Mark an instruction as not executable. You can use this to include comments in the script.
-  ■ Up arrow
Move an instruction up in the script
-  ■ Down arrow
Move an instruction down in the script

You can copy, paste, and delete instructions using the shortcut keys described in Table 5-1.

Understanding Repository

The repository structure, shown in Figure 5-5, displays the databases, their tables, and columns. Use Repository to drag and drop column definitions when you create filters in transformation instructions.

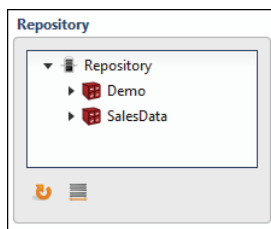




Figure 5-5 Repository structure

-  ■ Refresh the repository content.
-  ■ Get derived column definitions.
Displays a list of the derived columns in the selected database, as shown in Figure 5-6.

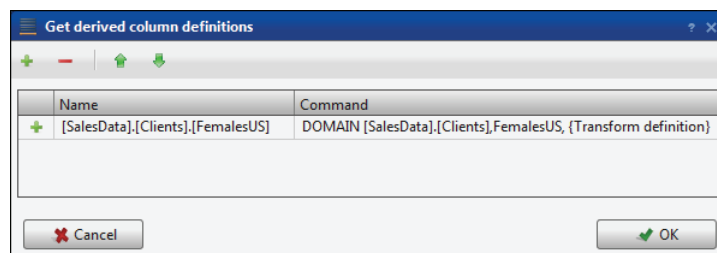


Figure 5-6 Getting a derived column definition

- Double-click a table column to explore the column properties and discrete values.

Typically, analysts create derived (calculated) columns during the data analysis. BIRT Loader administrators have to add these columns to the tables when refreshing the data during the load process. Otherwise, the definitions of these columns are lost, and must be recreated. To add the definitions to a script, copy instructions from the list and paste them to the script in Main file.

Using the text editor

You can invoke the text editor from any box requiring text entry. Double-click the text box and the text editor appears, as shown in Figure 5-7.

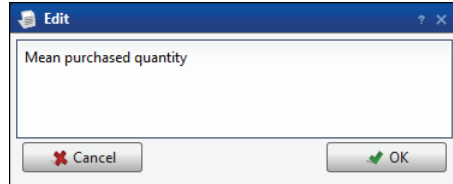


Figure 5-7 Using the text editor

Creating new columns

The instructions in this group create new dynamic columns in the database tables, using analytical functions to process the data. Once created the dynamic columns contain the formulas, used in their definition. The formulas can be edited and so the values of the columns get changed. The dynamic columns get populated with data at run time when analysis operations are performed. You have an option to make the columns permanent, and then the formulas are replaced with real data.

AGGREGATE

Creates an aggregated dynamic column. Aggregate allows you to group information from different tables. For example, you can group information about orders in a table that contains customer data in order to determine how many orders a customer has placed. Aggregation works with any table in the database as long as the tables are properly joined.

An Aggregate instruction must have the following syntax:

Syntax	<code>AGGREGATE [Target table] [Target column] [Source column] [Function] ([Filter])</code>
Parameters	<p>Target table Defines the name of the target table, for example [SalesData].[MonthlySales]. The definition contains the database name, followed by the table name.</p> <p>Target column Defines the name of the target column. Do not use the following characters: / \ ° * - accents, dieresis, ?, !, *, @, #, &, \$, o, ñ</p> <p>Source column Defines the source column for the aggregation. The full definition includes, the database name, the table name, and the column name.</p> <p>Function Defines the aggregation function. The supported functions are shown in Table 5-2.</p>

Table 5-2 Aggregation functions

Function	Returns
Average	The average value
Count	The record count
First	The minimum value of an alphabetically sorted range of records
Last	The maximum value of an alphabetically sorted range of records
Sum	The sum of the column value in all records
Mean (integer)	The mean value in Integer format
Mean (real)	The mean values in Real format
Minimum value	The minimum value of the selected range of records
Maximum value	The maximum value of the selected range of records
Standard deviation	A value indicating the average distance from the average
Square sum	The sum of squares of the distance between each data point and the line of best fit

Filter

Use Domain definition to define a filter.

Description

Type text that describes the new column. You can type the text in the Description text box, or double-click to open a text editor and type the text there.

How to create an aggregating column

- 1 In Main File, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, select AGGREGATE. Aggregate appears as shown in Figure 5-8.

The screenshot shows the 'Aggregate' dialog box with the following configuration:

- Table:** [SalesData] and [MonthlySales]
- Column:** Ave-Quantity
- Source column:** [SalesData], [DailyPurchases], [QUANTITY]
- Function:** Mean (Integer)
- Filter:** Click to select define a domain
- Description:** Mean purchased quantity

Figure 5-8 Creating an aggregating column

- 3 In Table, select the target database and table from the drop-down lists.
- 4 In Column, type the name of the target column.
- 5 In Source column, select the database, the table, and the column you want to use to create the new column.
- 6 In Function, select a function from the drop-down list.
- 7 In Filter, choose Click to select define a domain, to define a filter. Domain definition appears. For more details about how to define a domain see DOMAIN, later in this chapter.

- 8 In Description, type text that describes the new column.
- 9 Choose OK. The aggregate instruction appears in Main file, as shown in Figure 5-9.

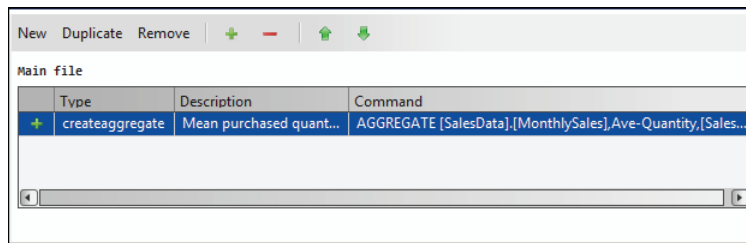


Figure 5-9 Creating a transformation instruction

- 10 Choose Save to save the instruction in the main file.

DECODE

Creates a dynamic decoding column. You can decode any column that contains up to 100 discrete values. For example, gender is usually encoded as M and F, or even as 0 and 1. BIRT Analytics makes it possible to change the name to something more recognizable, such as Male and Female.

A Decode instruction must have the following syntax:

Syntax `DECODE [Source column] [Target column] [File] [Code position]
[Decode position] [Encoding] [End line] [Delimiter] [Qualifier]`

Parameters **Source column**
Defines the source column name.

Target column
Defines the target column name. The column is added to the source table. Do not use the following characters:

/ \ ° ° - accents, dieresis, ?, !, *, @, #, &, \$, o, ñ

File
The name of the file containing decoding values. The file format must comply with the text file formats supported by BIRT Analytics Loader. Use File Upload in Data tab to upload the decoding file and make it available in BIRT Analytics Loader.

Code position
The position of a code in the file.

Decode position
The position of a decoding value in the file.

Encoding
Defines the file encoding.

End line
The end-of-line character. Select a character from the drop-down list.

Delimiter
Separates the values in the file. Select a delimiter from the list of supported characters.

Qualifier
Surrounds the values in the file. Select a qualifier from the list of supported characters.

How to create a decoding column

Before creating a new decoding column, you must create a text file with the decoding values and deploy the file to BIRT Analytics Loader.

- 1 In Main File, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, select Decode. Decode appears, as shown in Figure 5-10.

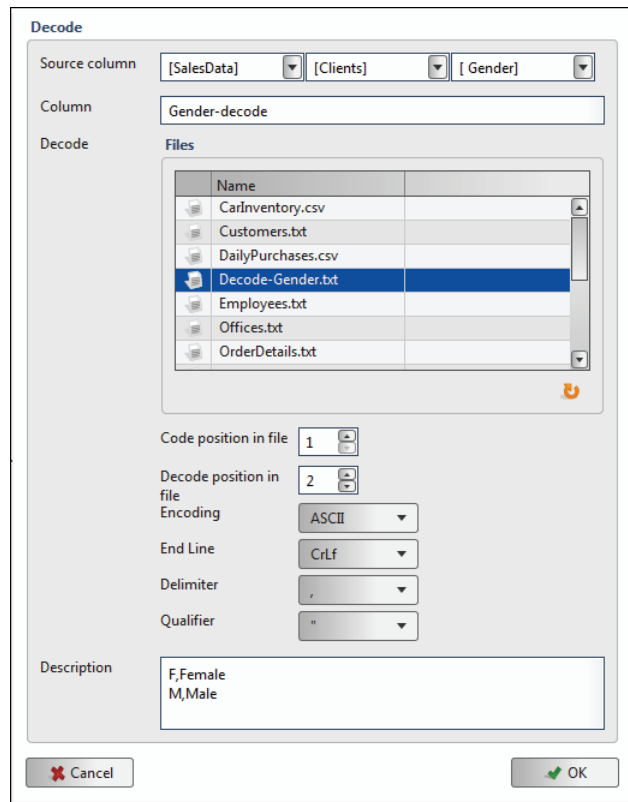


Figure 5-10 Creating a decoding column

- 3 In Source column, select a database name, a table name, and a column name from the drop-down lists.
- 4 In Column, type a name for the new column.
- 5 In Files, select a file with decoding values. In this example, decode-gender.txt file contains the following two comma-separated columns with the following coding and decoding values:
M, Male
F, Female
- 6 In Code position in file, type or select the number of the code column in the file. In this example, the position of the code is 1.
- 7 In Decode position in file, type or select the position of the decoding column. It is 2 in this example.
- 8 In End line, select the end line character used in the file from the drop-down list.
- 9 In Delimiter, select the delimiter used in the file from the drop-down-list.
- 10 In Qualifier, select the qualifier from the list.
- 11 In Description, type a description of the column, or double-click to open a text editor and type the text there.
- 12 In Decode, select OK. The Decode instruction appears in Main file, as shown in Figure 5-11.

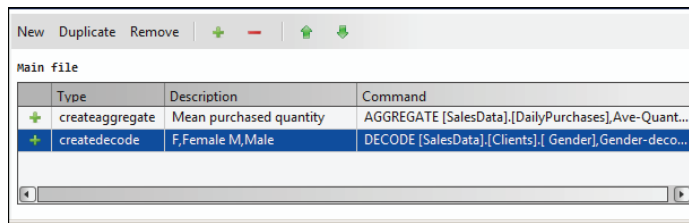


Figure 5-11 Creating a Decode instruction

13 Choose Save to save the project.

14 Validate the new column was created.

- 1 Choose Data Management, and then Explorer.
- 2 In Database, select your database, and expand the list of tables. The list of tables appears.
- 3 In Tables, double-click the table you added the column to. The list of columns appears. The new decode column, in this example Gender-decode, appears, as shown in Figure 5-12.

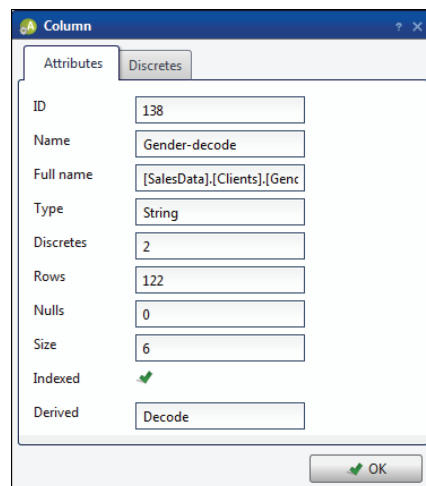


Figure 5-12 Validating the decode column

DOMAIN

A domain is a segment, or a set of records chosen for a specific purpose. A database is made up of records, and these records internally form groups that have similar features. For example, a company's customers can be female or male, but not both. For example, to specify the target audience for a marketing campaign, you can combine segments, such as females at age between 25 and 35, from Barcelona, and average purchasing power.

Domain can be defined as a simple query, or as a more complex set of segments. A Domain instruction must have the following syntax:

Syntax DOMAIN [Target table] [Target column] [Domain]

Parameters **Target table**
Defines a target table.








Target column
Defines a target column. Do not use the following characters:

/ \ ° ° - accents, dieresis, ?, !, *, @, #, &, \$, o, ñ

Domain

Defines a domain name. Specify a query that defines a domain, using any of the tools described in Table 5-3.

Table 5-3 Domain definition features

Icon	Tool	Description
	Simple query	Defines a simple query. Allows you to create a query, by selecting database columns, and operations from the database.
	New block	Defines a new block of segments. A segment may contain multiple queries. A domain can be build of multiple blocks. The blocks act as parenthesis and define the order of execution.
	Resolution	Changes the resolution of the whole domain or of a set of values entered in a block. For example, if you make a domain of records of people who are women and managers, and you want to see the households where the people who meet these conditions live, you can use this option to check the results in a dimension other than the source of the values.
	Inversion	Inverts the selected value or values. For example, if you use Invert on a domain that contains the records of people that are women and managers, you get the records of people who are not women and not managers.
	Discrete values	Selects the values that create the domain. Select a value from a column in the database.
	Sorting	Sorts the values in a column in ascending or descending order.
	Sample selection	Creates a sample of number of records.

You can find more information about domain features in *Using BIRT Analytics*.

How to create a domain

- 1 In Main File, choose New, and then Domain in the list of transformation instructions. Domain appears.
- 2 In Table, select the database, and the table from the drop-down list, or select the table in Repository and drag it to the column box.
- 3 In Column, type the new column name, as shown in Figure 5-13.

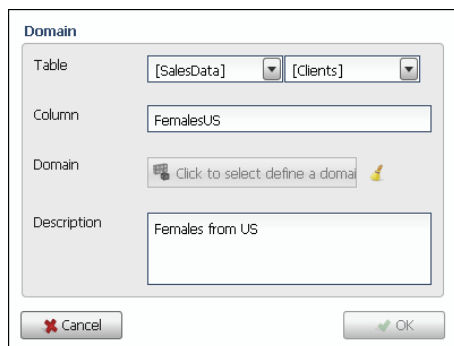


Figure 5-13 Creating a domain

- 4 In Domain, choose Click to select define a domain. Domain definition appears.
- 5 Create a new query, as shown in Figure 5-14.

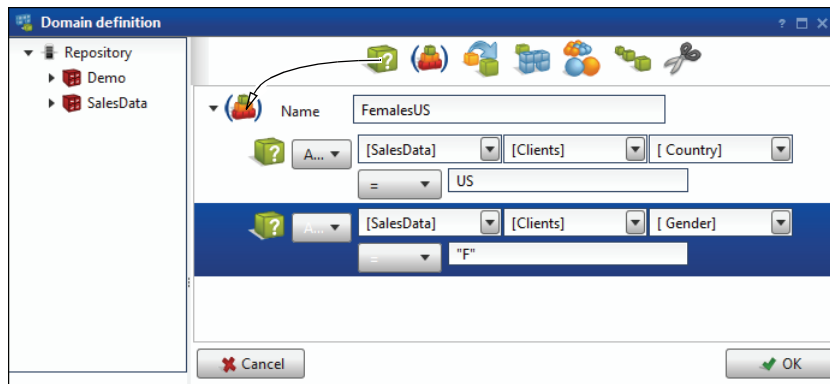


Figure 5-14 Defining a domain

- 1 In Name, type the name of the domain.
 - 2 Drag New query from the toolbar over New block, located on the left side of Name, as shown in Figure 5-14.
 - 3 In New query, select a database, a table, and a column, or drag a column from Repository.
 - 4 In New query, select the operator, for example =, from the drop-down list.
 - 5 Type a value, for example US.
 - 6 Drag a New query to the New block to add a second condition.
 - 7 Select the operator between the two conditions. If you wish to get the records for all women in US, you must select AND.
 - 8 Select a database, a table, and a column for the second condition.
 - 9 Select operator, and type or select a value for the comparison.
 - 10 In Domain definition, choose OK.
- 6 In Domain, choose OK. The new domain instruction appears in Main file, as shown in Figure 5-15.

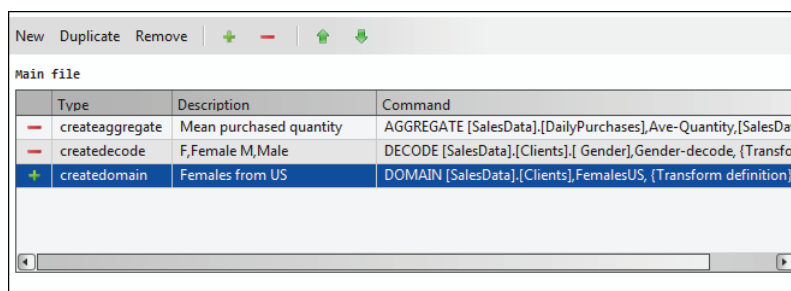


Figure 5-15 Creating a Domain instruction

- 7 Choose Save to save the project.
- 8 Validate the new column was created.
 - 1 Choose Data Management, and then Explorer.
 - 2 Expand the list of tables and columns in the Repository tree. The new Domain column, in this example FemalesUS, appears in the list. Select the column and choose View details to examine the new column properties and values, as shown in Figure 5-16.

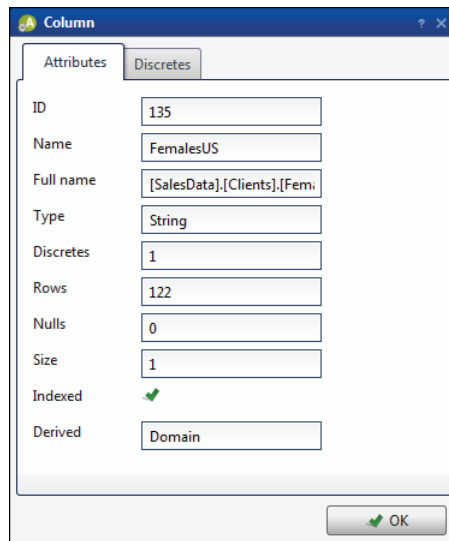


Figure 5-16 Validating the new domain column

EXPRESSION

Using the Expression instruction you can create calculated dynamic columns using one or two columns from a database. You can create simple expressions or complex expressions. For example, a simple expression concatenates strings by joining several columns. A complex expression uses mathematical operations such as logarithms. For expressions that generate a decimal value, you can specify precision up to six decimal places.

An Expression instruction must have the following syntax:

Syntax `EXPRESSION [Target table] [Target column] [Expression]`

Parameters **Target table**
Define the target table where the expression column is created.

Target column
Defines the name of the new column. Do not use the following characters:

/ \ ° ª - accents, dieresis, ?, !, *, @, #, &, \$, o, ñ

Expression
The expression used to create a column value. To create expressions you use operators, statements, and functions.

BIRT Analytics Loader supports the following operators described in Table 5-4.

Table 5-4 Supported operators and statements

Name	Description	Example
+	Concatenate two text columns.	<code>[Demo].[Customer].[Title]+[Demo].[Customer].[Surname]</code>
+	Concatenate Numeric columns or strings.	<code>STRING([CustomerID])+5+[DaysSinceLastOrder]-1</code>
+, -, *, /	Mathematical operators for adding, subtracting, multiplying, or dividing numeric columns.	<code>STRING([Demo].[Customer].[Cust_ID]+[DEMO].[Customer].[Household_ID])</code>

Table 5-4 Supported operators and statements

Name	Description	Example
GT[>], LT [<], GE[>=], LE[<=], EQ [=], NE[<>], AND, OR, NOT	Assign Boolean values to columns, based on the conditions.	[Demo] . [Customer] . [Edats] EQ "Adults" >0 or [Demo] . [Customer] . [Edats] EQ "Joves" >0
Constants	Assign a constant to a table column.	[Demo] . [Customer] . [Household_ID] =1
Switch statement	Assign a value to a column depending on a condition.	SWITCH ([Demo] . [Customer] . [Gender]) CASE "M" : "MALE" , CASE "F" : "FEMALE" SWITCHEND
Boolean If statement	Return a value depending on a condition. The condition can be simple, or more complex, and the return value can be any type, for example yes or no.	If ([DEMO] . [Order] . [OrderDate] > [Demo] . [Customer] . [DOB] , 1 , 0)

Table 5-5 presents the complete list of the functions. For more information about these functions, see *Using BIRT Analytics*.

Table 5-5 Supported functions

Name and syntax	Description
ABS ([db].[table].[column])	Returns the absolute value of a numeric column, regardless of sign.
AGE ([db].[table].[column], date)	Returns the age of each value on a specified date.
ALLMONTHS ([db].[table].[column])	Returns the year and the month of each one of the values of a date column.
ALLQUARTERS ([db].[table].[column])	Returns the year and the quarter of each one of the values of a date column.
CEIL ([db].[table].[column])	Given a continuous numeric column, it returns the next integer value. For example, given the value 3.6, the CEIL function returns 4.
CHR ([db].[table].[column])	Given an integer numeric column, it returns the associated ASCII (American Standard Code for Information Interchange) value.
CODE ([db].[table].[column])	Returns the ASCII code for the first value of the text string. It only works for ASCII string columns, not for Unicode columns.
COUNT ([db].[table].[column])	Returns the number of records in a table.
DATE ([db].[table].[column])	Enters a column containing date data as text strings, date-and-time format or the data for a constant date.
DATETIME ([db].[table].[column])	Enters a column containing date data as text strings, or unicode, or a constant date.
DAY ([db].[table].[column])	Returns the part for the day in a date column.
DAYADD ([db].[table].[column], [db].[table].[column2])	Adds or removes the number of days required in a date column. The column returned is a date column.

(continues)

Table 5-5 Supported functions (continued)

Name and syntax	Description
DAYSTO ([db].[table].[column], date)	Returns the number of days between each value and a specified date.
DISTINCT ([db].[table].[column])	Returns the unique value or values in a column. For example, in a [Demo].[Customer].[Surname] column, it returns each surname stored in the column only once.
EXP ([db].[table].[column])	Returns the exponential value of each of the values of a numeric column.
FLOOR ([db].[table].[column])	Returns the largest integer number less than or equal to the array.
HOUR ([db].[table].[column])	Returns the time of each date.
IF (A, B, C)	Returns a conditional function. A is the condition to be applied, B is the value to return if the condition is true, C is the value to return if the condition is false.
INSTR ([db].[table].[column], "string")	Returns an integer that specifies the initial position of the first appearance of a string in another one. First value is 0.
INT ([db].[table].[column])	Returns the integer number that forms part of the value in a column, or converts the column that contains the numbers into a numeric column.
ISNULL ([db].[table].[column])	Returns true for null values and false for not-null values.
ISNOTNULL ([db].[table].[column])	Returns true for not-null values and false for null values.
KURT ([db].[table].[column])	Returns the kurtosis of a column, which represents how values are distributed around the mean.
LCASE ([db].[table].[column])	Returns a lowercase text string, turning uppercase characters into lowercase.
LEFT ([db].[table].[column], n)	Returns the left part of a string of characters, where <i>n</i> is the number of characters to return.
LEN ([db].[table].[column])	Returns the number of characters in a text column, returning a number for each value in the column.
LOG ([db].[table].[column])	Returns the natural logarithm (logarithm in base e) of the values of the column.
LOG10 ([db].[table].[column])	Returns the decimal logarithm (logarithm in base 10) of the values of the column.
LONGINT ([db].[table].[column])	Returns the long integer part of a column, or converts the column that contains a number into a numeric column.
LTRIM ([db].[table].[column])	Returns a string that is a copy of a string with no initial spaces (deletes any initial spaces).
MAX ([db].[table].[column])	Returns the maximum value of a column. The column used must always be numeric (continuous or discrete).
MEAN ([db].[table].[column])	Returns the average value of a numeric column (continuous or discrete).
MEDIAN ([db].[table].[column])	Returns the value that divides the values of a column into two equal parts.

Table 5-5 Supported functions (continued)

Name and syntax	Description
MID ([db].[table].[column], number1, number2)	Returns a string that contains a specified number of characters. Number1 indicates the position where the string to be returned starts and Number2 indicates the number of characters to return (optional). If Number1 is greater than the number of characters in the initial string, it returns a 0-length string.
MIN ([db].[table].[column])	Returns the smallest value of the parameters of a numeric column (continuous or discrete).
MINUTE ([db].[table].[column])	Returns the minutes part of a date and/or time column.
MODE ([db].[table].[column])	Returns the value that is most repeated in the values of a column (the most frequent value).
MONTH ([db].[table].[column])	Returns the month part of a date column.
NOW ()	Returns the current date and time in the engine's default format.
POWER ([db].[table].[column], number)	Returns all the values of the column to the specified second parameter (number).
PROPER ([db].[table].[column])	Returns a new column in which all records begin with an uppercase letter and the rest are in lowercase.
PROPERSENTENCE ([db].[table].[column])	Returns a new column in which the first letter of each word in the description of the variables is in uppercase.
QUARTER ([db].[table].[column])	Returns the quarter of the year for each of the records in the date column.
RAND (n)	Returns a column based on random (<i>n</i>). For example, RAND (120) in the Customers table returns a column in which customers are randomly divided into 120 different groups.
REAL ([db].[table].[column])	Returns the real value of a column. The source column must be a numeric column, a text string containing text, or a string representing a numeric column.
REPLACE ([db].[table].[column], pattern, replacement)	Replaces the pattern text with the replacement text. Both pattern and replacement can be columns or text
RIGHT ([db].[table].[column],n)	Returns the right part of a string of characters where n is the number of characters to return.
ROW ([db].[table].[column])	Returns a unique value, starting from 0, for each of the existing records in the column used to create the expression.
RTRIM ([db].[table].[column])	Returns a text string, an exact copy of the specified string without trailing spaces.
SECOND ([db].[table].[column])	Returns the seconds part of a date or time column.
SECSTO ([db].[table].[column], [db].[table].[column])	Returns the seconds elapsed between two columns with date or time format.
SGN ([db].[table].[column])	Returns the values grouped by -1 for negative values, 0 for invalid values, because not entered, for example, and +1 for positive values.
SKEW ([db].[table].[column])	Returns the skewness value of the values of a column with respect to the mean value.

(continues)

Table 5-5 Supported functions (continued)

Name and syntax	Description
SQRT ([db].[table].[column])	Returns the square root of <i>n</i> , where <i>n</i> is the mean of the values of the column.
STDEV ([db].[table].[column])	Returns the standard deviation of the values of the column, which is the square root of the variance.
STRING ([db].[table].[column])	Returns a text string that represents any other type of data, except Unicode.
STRIP ([db].[table].[column])	Returns the entered column deleting any spaces.
SUM ([db].[table].[column])	Returns a new column with the cumulative sum of the values of the original column.
SUMSQ ([db].[table].[column])	Returns a new column with the cumulative sum of each of the values of the original column squared.
TIME ([db].[table].[column])	Returns the time of a string, date-and-time, or time column.
TODAY ()	Returns current date in engine's format.
TRIM ([db].[table].[column])	Returns a text string that contains the copy of the specified string with no spaces either at the beginning or the end of the string.
UCASE ([db].[table].[column])	Returns a new column with text in uppercase.
VAL ([db].[table].[column])	Returns a Unicode string that represents any other type of data.
WEEKDAY ([db].[table].[column])	Depending on the locale, returns the day of the week for each of the values. Default locale is en_US. For example, in the en_US locale: 1=Sunday. In es_ES locale: 1=Monday.
YEAR ([db].[table].[column])	Returns the year of a date or date-and-time column.

How to create an expression

This example creates a column that displays the total dollar amount of car inventory.

- 1 In Main file, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, select Expression. Expression appears, as shown in Figure 5-17.

Figure 5-17 Creating an expression

- 3 In Table, select a target database and a table.
- 4 In Column, type the name of the new column.
- 5 In Expression, type the expression.

- 6 In Description, type the description of the column.
- 7 In Expression, choose OK. The Expression instruction appears in Main file, as shown in Figure 5-18.

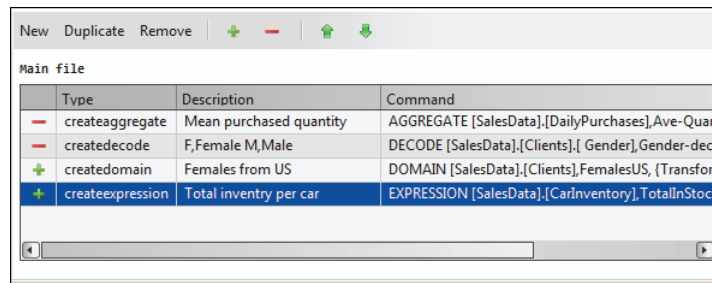


Figure 5-18 Creating an Expression instruction

- 8 Choose Save to save the project.
- 9 Validate that the new column was created in Explorer in Data Management.

INUMERIC

Using numeric ranges, you can create a dynamic column that groups values into numeric ranges. A numeric range must be based on a numeric column. Numeric ranges are useful for simplifying data. For example:

- There is a column in the database containing profit per order. Depending on the type of analysis you want to do, it might be useful to group these figures into five intervals such as very high, high, medium, low, and very low.
- Another very common case is grouping individuals into age ranges. These ranges tend to be very different depending on the type of analysis you want to do, and so being able to configure the ranges manually gives you the flexibility you need.

An INumeric instruction must have the following syntax:

Syntax INUMERIC [Source column] [Target column] [Bands] ([Filter])

Parameters Source column

Defines a database, table, and column for a source for the numeric ranges. The column must be of numeric type.

Target column

Defines the name of the target column. The column is added to the source table.

Bands

Defines the ranges in the following format:

[Lower range number] | [Upper range number] | [Range Name]

Each range must be on a separate line. The lower and upper numbers define the numbers to include in the range. For example, if you want to define three age groups in households, you use the following definition:

```
1 | 30 | Young
31 | 55 | Middle-age
56 | 105 | Old
```

Filter

Defines or uses an existing domain to filter the data set in the table.

How to create numeric ranges

- 1 In Main file, choose New. The list of transformation instructions appears.

- 2 In the list of transformation instructions, select INUMERIC. Numeric ranges appears, as shown in Figure 5-19.

Numeric ranges

Source column: [SalesData] [Employees] [SAL]

Column: SalaryRanges

Ranges: 0|1000|low, 1001|3000|middle, 3001|5000|high

Filter: Click to select define a domain

Description: Salary ranges: low, middle and high

Cancel OK

Figure 5-19 Creating numeric ranges

- 3 In Source column, select a database, a table, and a column that will be used as a source for the numeric range column. The column must be numeric.
- 4 In Column, type the name of the new numeric range column.
- 5 In Ranges, define the ranges and their names. Double-click in the box to open the text editor and type the text.
- 6 In Numeric ranges, choose OK. The INumeric instruction appears in Main file, as shown in Figure 5-20.

New Duplicate Remove + - ↑ ↓

Main file

Type	Description	Command
createaggregate	Mean purchased quantity	AGGREGATE [SalesData].[DailyPurchas
createexpression	Total inventory per car	EXPRESSION [SalesData].[CarInventory
createnumeric	Salary ranges: low, middle a...	INUMERIC [SalesData].[Employees].[SA

Figure 5-20 Creating an INumeric instruction

- 7 Choose Save to save the project.
- 8 Run the instruction.



- 1 In Main file, select each instruction and then, -, to make it inactive, as shown in Figure 5-20.
- 2 In Data—Options, deselect Load, as shown in Figure 5-21.

Options

☐ Load ☒ Transform Notify 0 Sample 0

Figure 5-21 Deselecting Load

- 3 Choose Run. As the project runs, messages describing each stage of the project appear in Running project, as shown in Figure 5-22.

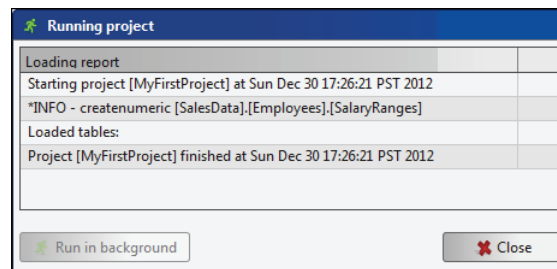


Figure 5-22 Running the INumeric instruction

- 9 Validate that the new column was created in Explorer in Data Management.

PARAMETRIC

Parametric columns consist of query-based values. These queries are formulated with values or columns belonging to any table in the database as long as the tables are properly joined. The resulting dynamic column is created in the table indicated during the process, so the results are very different depending on the resolution marked by the selected table.

For example, in the Demo database, you can determine sales of a particular product. For the Customers table, the query returns the number of customers who bought one or more quantities of the product, while for the Products table the query returns how many X products have been sold. The number of products sold is probably greater than the number of customers who have bought them, because one customer may buy more than one unit of a product, but one unit of a product cannot be sold to more than one customer.

A parametric column is not displayed in the data explorer, since the same record may meet the condition of several values in the parametric column.

A Parametric instruction must have the following syntax:

Syntax `PARAMETRIC [Target table] [Target column] [Bins]`
Parameters **Target table**
 Defines a database, and table.

Target column
 Defines the name of the target column.

Bins
 A set of data values that fall in a given interval. Define a bin name and a query that fills the bin with data. To define the query, use Domain.

How to create a parametric column

The example creates a parametric column, which groups the customers into two bins, customers from California, and customers with credit limit greater than \$100K.

- 1 In Main file, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, choose Parametric. Parametric appears, as shown in Figure 5-23.

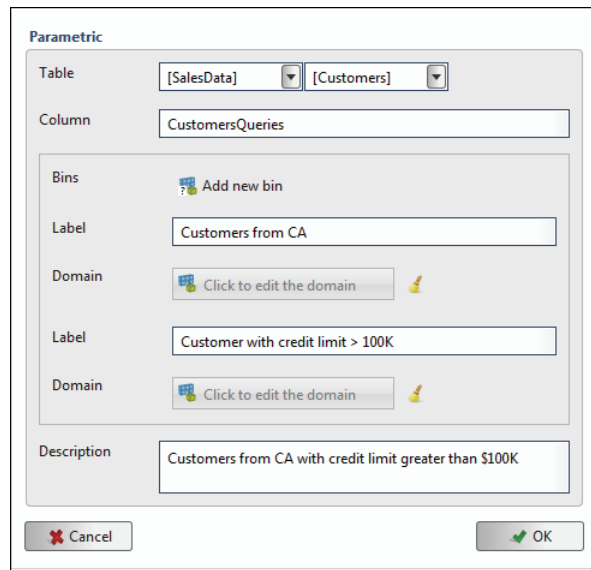


Figure 5-23 Creating a parametric column

- 3 In Table, select the target database and table.
- 4 In Column, type the name of the new column.
- 5 Define a new bin.
 - 1 In Bins, select Add new bin. The new bin definition appears.
 - 2 In Label, type the name of the new bin, for example Customers from CA.
 - 3 Choose Domain to define the query. Domain definition appears, as shown in Figure 5-24.

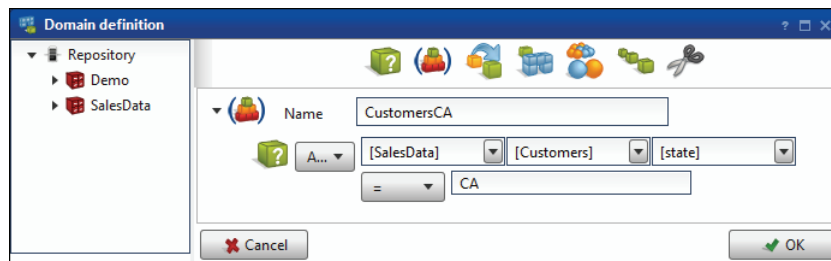


Figure 5-24 Creating the CustomersCA domain

- 4 In Domain definition, define the query and choose OK.
- 5 In Parametric, choose Add new bin again, to define another bin.
- 6 Define a label for the domain, and choose Domain to define the query. Domain definition appears again, as shown in Figure 5-25.

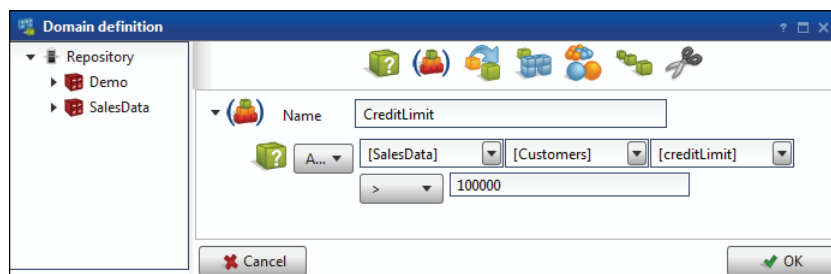


Figure 5-25 Creating the CreditLimit domain

- 7 In Domain definition, choose OK.
- 6 In Parametric, choose OK. The Parametric instruction appears, as shown in Figure 5-26.

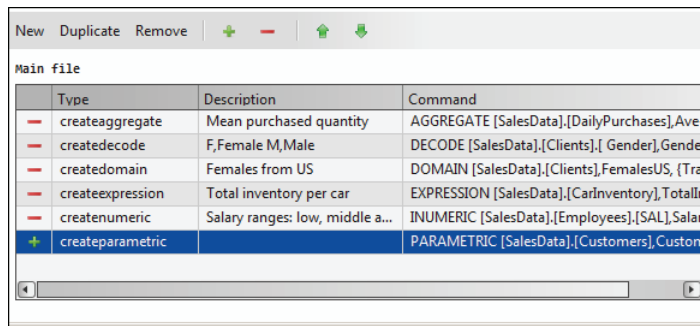


Figure 5-26 Creating a Parametric instruction

- 7 Choose Save, to save the project.
- 8 Run the instruction.



- 1 In Main file, select each instruction and then, choose - to make it inactive, as shown in Figure 5-20.
- 2 In Data—Options, deselect Load, as shown in Figure 5-27.

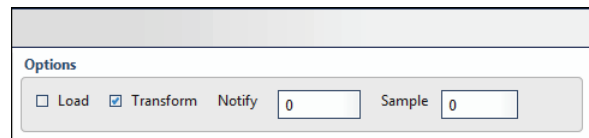


Figure 5-27 Deselecting Load

- 3 Choose Run. As the project runs, messages describing each stage of the project appear in Running project, as shown in Figure 5-28.

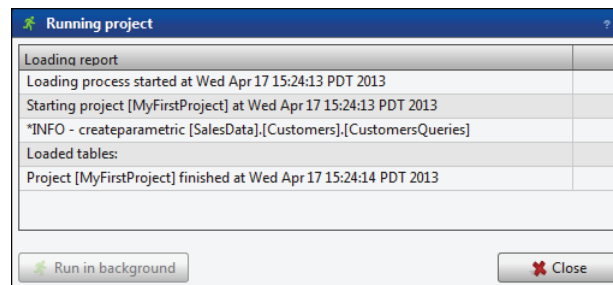


Figure 5-28 Running a Parametric instruction

- 9 Validate that the new column was created in Explorer in Data Management.

QUANTILE

A quantile range dynamic column is created from a numeric column. QUANTILE creates groups that contain the same number of values. These groups are created by sorting the values in ascending order so that the first range contains the smallest values and the last range contains the largest values. For example, you can identify customers that generate the greatest profit.

A Quantile instruction must have the following syntax:

- Syntax** QUANTILE [Source column] [Target column] [Number of parts] ([Filter])
- Parameters** **Source column**
Defines a numeric column.

Target column

Defines the name of the new target column.

Number of parts

Defines the number of parts the data must be grouped into.

Filter

Defines a domain, if the data requires filtering.

How to create a quantile range column

The following example describes how to create a quantile column that creates four groups that span the range of credit limit values found in a set of customer records. Table 5-6 lists the resulting quantile ranges and the number of customer records in each range.

Table 5-6 Quantile ranges

#	Credit limit range	Number of records
1	0-41900	31
2	43300-77000	31
3	77000-95100	30
4	95400-222600	30

- 1 In Main file, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, choose Quantile. Quantile appears, as shown in Figure 5-29.

The screenshot shows the 'Quantile' dialog box. It contains the following fields and controls:

- Source column:** Three dropdown menus showing '[SalesData]', '[Customers]', and '[creditLimit]'.
- Column:** A text box containing 'CreditLimitQuant'.
- Number of parts:** A numeric input field with the value '4' and increment/decrement buttons.
- Filter:** A button labeled 'Click to select define a domain' with a small icon.
- Description:** An empty text box.
- Buttons:** 'Cancel' (with a red X icon) and 'OK' (with a green checkmark icon) at the bottom.

Figure 5-29 Creating a quantile column

- 3 In Source column, select a database, a table, and a numeric column.
- 4 In Column, type the name of the new column.
- 5 In Number of parts, provide the number of quantiles, groups, or ranges that the column will contain.
- 6 In Filter, select Click to select define a domain, if you need to filter the data.
- 7 In Quantile, choose OK. The Quantile instruction appears in Main file, as shown in Figure 5-30.

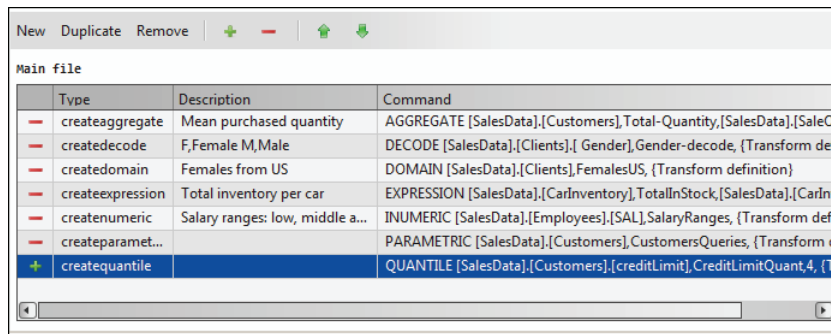


Figure 5-30 Creating a Quantile instruction

8 Save the project.

9 Run the instruction.



1 In Main file, select each instruction and then, choose - to make it inactive, as shown in Figure 5-20.

2 In Data—Options, deselect Load, as shown in Figure 5-31.

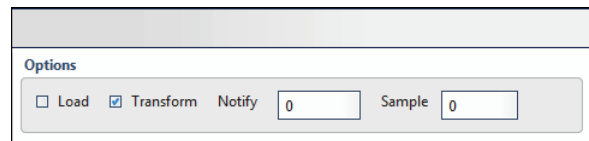


Figure 5-31 Deselecting Load

3 Choose Run. As the project runs, messages describing each stage of the project appear in Running project, as shown in Figure 5-32.

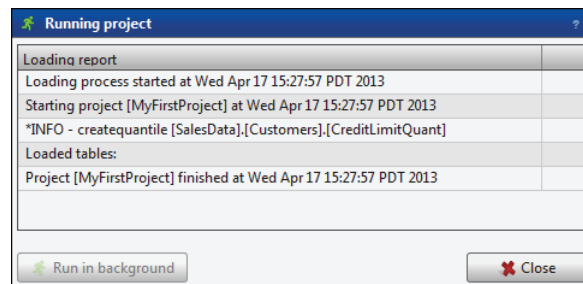


Figure 5-32 Running a Quantile instruction

RANKING

Supports creating a new column with the rank for each subset of records relating to the same key. Ranking allows you to rank a table based on a specific column. For each subset of records associated with the same key in a table, a sequence number is assigned according to the relative value of a given column. Ranking is used, for example, when trying to identify the first products purchased by each customer, or the most recent product purchased.

Ranking creates a column with an integer value, which ranks records in the same table. The instruction syntax is:

Syntax RANKING [Key to rank] [Column to use] [Column name] [Sorting order] ([Force sequence] [Filter])

Parameters **Key to rank**

Defines a column that represents the key to be ranked. The column must be an index key in a table.

Column to use

Defines the column to use when creating the ranking numbers.

Column name

Defines the name of the new ranking column.

Sorting direction

Defines the sorting direction, ASC or DESC.

Force sequence

Defines the ranking sequence. In cases where two or more records have exactly the same value, checking Force sequence ensures that each record is allocated a different rank. If you do not check Force sequence, records with the same value have the same rank.

Filter

Defines a domain in case you want to filter the data.

How to create a ranking column

- 1 In Main file, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, choose Ranking. Ranking appears, as shown in Figure 5-33.

The screenshot shows the 'Ranking' configuration window. It includes several dropdown menus for selecting data sources and columns, a text field for the column name, a dropdown for sorting direction, a checkbox for 'Force sequence', a button for the filter, and a text area for a description. The 'OK' button is highlighted in green.

Figure 5-33 Creating a ranking column

- 3 In Key to rank, select a database, a table, and a column you use as a key for ranking.
- 4 In Field to use, select a database, a table, and a column you use to determine the ranking values.
- 5 In Name, type the name of the new ranking column.
- 6 In Direction, select the table-sorting direction from the drop-down list.
- 7 Select Force sequence.
- 8 In Filter, choose Click to select define a domain, if you wish to filter the data.
- 9 In Description, type a short description of the column.
- 10 Choose OK. The Ranking instruction appears in Main file, as shown in Figure 5-34.

New Duplicate Remove + - ↕			
Main file			
	Type	Description	Command
-	createaggregate	Mean purchased quantity	AGGREGATE [SalesData].[Customers],Total-Quantity,[SalesData].[SaleOrders],[QU...
-	createdecode	F,Female M,Male	DECODE [SalesData].[Clients],[Gender],Gender-decode, {Transform definition}
-	createdomain	Females from US	DOMAIN [SalesData].[Clients],FemalesUS, {Transform definition}
-	createexpression	Total inventory per car	EXPRESSION [SalesData].[CarInventory],TotalInStock,[SalesData].[CarInventory],[...
-	createnumeric	Salary ranges: low, middle a...	INUMERIC [SalesData].[Employees],[SAL],SalaryRanges, {Transform definition}, {T...
-	createparamet...		PARAMETRIC [SalesData].[Customers],CustomersQueries, {Transform definition}
-	createquantile		QUANTILE [SalesData].[Customers],[creditLimit],CreditLimitQuant,4, {Transform ...
+	createreanking	Rank customers, depending...	RANKING [SalesData].[Customers],[customerNumber],[SalesData].[SaleOrders],[o...

Figure 5-34 Creating a Ranking instruction

11 Save the project.

12 Run the instruction.



- 1 In Main file, select each instruction and then, choose - to make it inactive, as shown in Figure 5-20.
- 2 In Data—Options, deselect Load, as shown in Figure 5-35.

Options

☐ Load
 ☒ Transform
 Notify
 Sample

Figure 5-35 Deselecting Load

- 3 Choose Run. The project runs and a confirmation message appears.

Managing data objects

The instructions in this group work with data objects such as databases, tables, and columns.

DROPCOLUMN

Removes a column from a table. The syntax is:

Syntax DROPCOLUMN [Column name]

Parameter **Column name**
Define the name of the column you want to drop from a table, as shown in Figure 5-36.

Drop column

Column

[SalesData] [DailyPurchases] [aaa]

Description

Remove a column from a table

Cancel

OK

Figure 5-36 Removing a column from a table

DROPDATABASE

Removes a database from the repository. The syntax is:

Syntax DROPDATABASE [Database name]

Parameter Database name

Defines the name of the database you wish to remove from the repository, as shown in Figure 5-37.

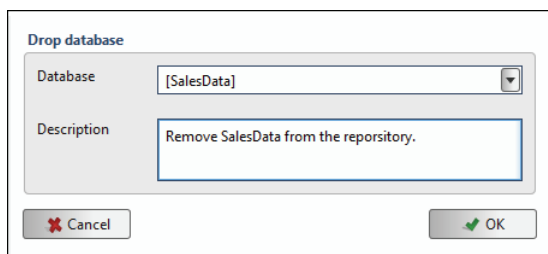
A dialog box titled "Drop database". It contains a "Database" field with a dropdown menu showing "[SalesData]". Below it is a "Description" text area containing the text "Remove SalesData from the repository.". At the bottom are two buttons: "Cancel" with a red X icon and "OK" with a green checkmark icon.

Figure 5-37 Removing a database from the repository

DROPTABLE

Removes a table from the repository. The syntax is:

Syntax DROPTABLE [Table name]

Parameter Table name

Defines the name of the table you wish to remove from a database, as shown in Figure 5-38.

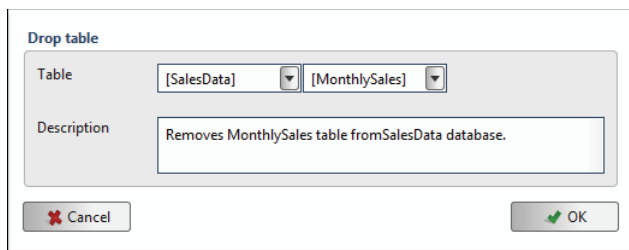
A dialog box titled "Drop table". It contains a "Table" field with two dropdown menus. The first dropdown shows "[SalesData]" and the second shows "[MonthlySales]". Below it is a "Description" text area containing the text "Removes MonthlySales table fromSalesData database.". At the bottom are two buttons: "Cancel" with a red X icon and "OK" with a green checkmark icon.

Figure 5-38 Removing a table from a database

DROPLINK

Removes a link between two tables. Use Droplink to delete existing relations between columns in different tables. If the links have not been created or have been created incorrectly, it is not possible to work with various levels of resolution in the analysis. The syntax is:

Syntax DROPLINK [From column name] [To column name]

Parameters From column name

Defines the name of the column from which the join is initiated.

To column name

Defines the name of the column to which the link is created. Figure 5-39 shows Drop link.

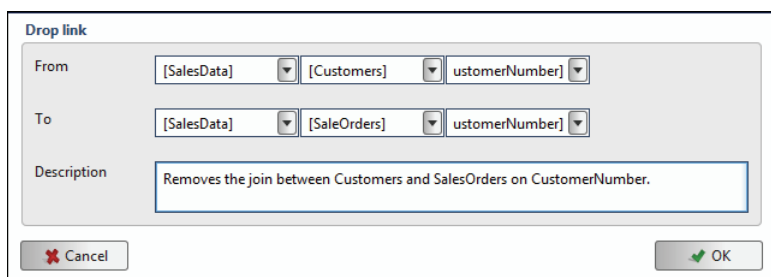
A dialog box titled "Drop link". It contains two rows of dropdown menus. The "From" row has three dropdowns: "[SalesData]", "[Customers]", and "ustomerNumber". The "To" row has three dropdowns: "[SalesData]", "[SaleOrders]", and "ustomerNumber". Below these is a "Description" text area containing the text "Removes the join between Customers and SalesOrders on CustomerNumber.". At the bottom are two buttons: "Cancel" with a red X icon and "OK" with a green checkmark icon.

Figure 5-39 Removing a link between tables

LINK

Creates relations between columns in different tables. Links can also be created using the Links option in Data Management. If the links have not been created or have been created incorrectly, it is not possible to work with various levels of resolution in the analysis.

Creates a link between two tables. The syntax is:

- Syntax

LINK [From column name] [To column name]
- Parameters

From column name
Defines the name of the column from which the join is initiated.

To column name
Defines the name of the column to which the link is created.

How to create a link between two tables

The example creates a link between two tables, Customers and SalesData on CustomerNumber.

- 1 In Main file, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, choose Link. Link appears, as shown in Figure 5-40.

Link

From

[SalesData][Customers][customerNumber]

To

[SalesData][SaleOrders][customerNumber]

Description

Create link between Customers and SaleOrders on CustomerNumber.

Cancel

OK

Figure 5-40 Creating a link between two tables

- 3 In From, define a database, a table, and a column the link initiates.
- 4 In To, define a database, a table, and a column to which the link points.
- 5 In Link, choose OK. The link instruction appears in Main file, as shown in Figure 5-41.

New Duplicate Remove + - Up Down

Main file

	Type	Description	Command
-	createaggregate	Average purchased quantity	AGGREGATE [SalesData].[DailyPurchases],Total
-	createparametric		PARAMETRIC [SalesData].[Customers],Customer
+	createlink		LINK [SalesData].[Customers],[customerNumber]

Figure 5-41 Creating a Link instruction

- 6 Save the project.
- 7 Run the instruction.



- 1 In Main file, select each instruction and then, choose - to make it inactive, as shown in Figure 5-20.
- 2 In Data—Options, deselect Load, as shown in Figure 5-42.

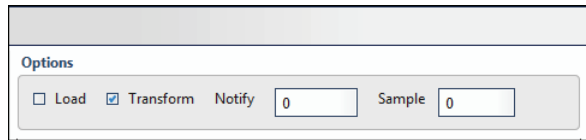


Figure 5-42 Deselecting Load

- 3 Choose Run. As the project runs, messages describing each stage of the project appear in Running project.

MAKEPERMANENT

The columns created by transformation functions are dynamic by nature and contain the formulas used for their definition, instead of real data. MAKEPERMANENT makes a dynamic column permanent, which means the column will be filled with data. The syntax is:

Syntax MAKEPERMANENT [Column name]

Parameter **Column name**

Defines the column name you wish to make permanent, as shown in Figure 5-43. Provide the full column definition in the following format:

[Database] [Table] [Column]

Note that the dynamic columns do not appear in the column's drop-down list. You must type the column name, surrounded by square brackets.

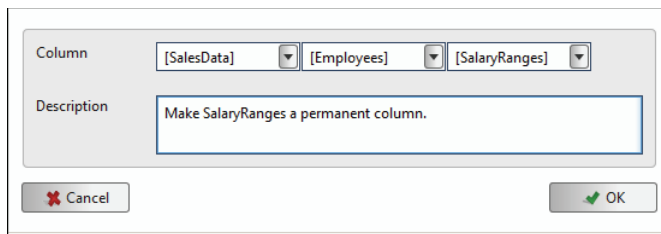


Figure 5-43 Making a column permanent

You can validate if the column is converted to permanent in Explorer. Select the column properties and check its Derived property. The property should be Not Derived.

For example, the [SalaryRanges] column is a dynamic column, created using the INUMERIC transformation instruction. The column's Derived property is numeric range before it is made permanent, as shown in Figure 5-44.

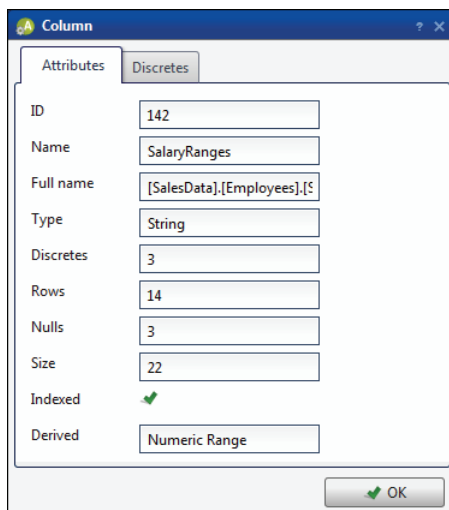


Figure 5-44 Dynamic column status

After the transformation the column status is changed to Not derived, as shown in Figure 5-45.

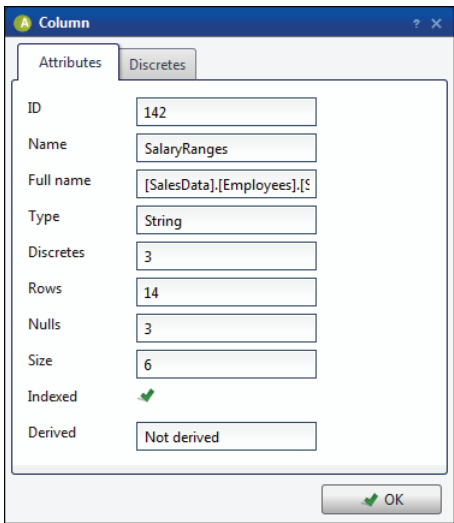


Figure 5-45 Permanent column status

RENAME

Renames a database object, such as a database, a table, or a column. The syntax is:

Syntax RENAME [Old name] [New name]

Parameters **Old name**
Defines a current name of a database, a table, or a column in the following format:
[Database] . [Table] . [Column]

New name
Defines a new name of a database, a table, or a column in the following format:
[Database] . [Table] . [Column]

Rename is shown in Figure 5-46.

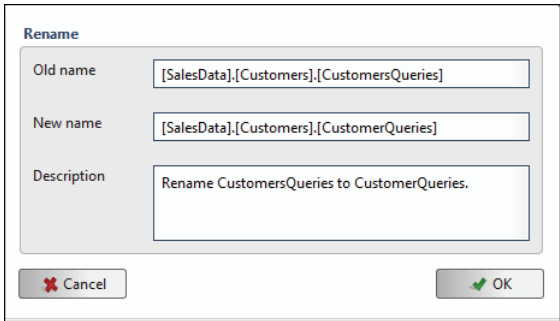


Figure 5-46 Renaming a column

Note, you must synchronize the database and set the privileges after you run the Rename instruction in order to see the database changes.

Miscellaneous instructions

The instructions in this group have different purposes.

EXPORT

Exports a domain to a table or a file. The syntax is:

Syntax EXPORT [Export target] [Target name] [Query resolution table]
[Query columns] [Filter]

Parameters **Export target and Target name**

Defines the object to where the data is exported. The export target can be:

- **Analytical database and table**
Defines new or existing database and table names.
- **CSV file**
Defines a CSV file name. Columns are delimited by the pipe (|) character. The file is stored in:

C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium

- **Fixed-length file**
Defines a file name. The columns have fixed length and the file is saved in:

C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium

Query resolution table

Defines the table you are exporting data from.

Query columns

Defines the query columns. Each column must be defined on a separate line, using the full name in the following format:

[Database] . [Table] . [Column]

To export the whole table, do not define columns.

Filter

Defines a domain you are using to filter the data. Filter is a required parameter.

How to export to a database

You have to choose the database where you want the new table to be created, the name of the new table, and the columns to be created in the destination table.

- 1 In Main file, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, choose Export. Export appears, as shown in Figure 5-47.

The screenshot shows the 'Export' dialog box. Under 'Export target', 'Table' is selected. The 'Query' section has 'Resolution table' with '[SalesData]' and '[Customers]' chosen. A text box lists columns: '[SalesData].[Customers].[customerNumber]', '[SalesData].[Customers].[city]', '[SalesData].[Customers].[state]', and '[SalesData].[Customers].[country]'. Below this is a 'Click to edit the domain' button. The 'Description' field says 'Export Customers with credit limit > \$100K'. 'Cancel' and 'OK' buttons are at the bottom.

Figure 5-47 Exporting data to a table

- 3 In Export target, select Table.
- 4 In Export target, type a new database name, or select an existing database name from the drop-down list.
- 5 In Export target, type a new table name, or select an existing table name from the drop-down list.
- 6 In Resolution table, select the table from which you want to export data.
- 7 In Enter columns, type the name of the columns, using the required format. Leave it empty if you wish to export the whole table. You can double-click in the box to open a text editor and type the column list.
- 8 Choose Click to edit the domain to define a domain that would filter the data. Defining a domain is required.
- 9 In Description, type a short description of the instruction's purpose.
- 10 In Export, choose OK to define the Export instruction.
- 11 Save the project.
- 12 Run the Export instruction.
- 13 To validate the export, open Explorer. Refresh the database. The new table appears, as shown in Figure 5-48.

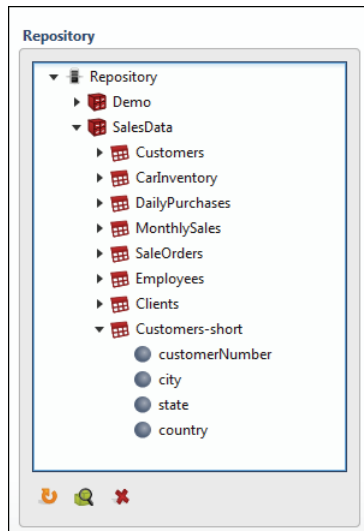


Figure 5-48 Validating an Export instruction

How to export data to a text file

- 1 In Main file, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, choose Export. Export appears, as shown in Figure 5-49.

Figure 5-49 Exporting data to a CSV file

- 3 In Export target, select CSV file.
- 4 In Export target, type the name of the CSV file.
- 5 In Resolution table, select the table from which you want to export data.
- 6 In Enter columns, type the name of the columns, using the required format. Leave it empty if you wish to export the whole table.
- 7 Choose Click to edit the domain to define a domain that would filter the data. Defining a domain is required.
- 8 In Description, type a short description of the instruction's purpose.
- 9 In Export, choose OK to define the Export instruction.
- 10 Save the project.
- 11 Run the Export instruction. The exported file is saved on the machine where BIRT Loader is installed. The file is located in:

C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium

The file format uses a pipe (|) as a column character delimiter, and CRLF as the end-of-line character, as shown in Listing 5-1.

Listing 5-1 Exported CSV file

```
114 |Melbourne|Victoria|Australia
119 |Nantes||France
124 |San Rafael|CA|USA
131 |NYC|NY|USA
141 |Madrid||Spain
146 |Lyon||France
```

How to export to a fixed-length file

- 1 In Main File, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, choose Export. Export appears, as shown in Figure 5-50.

Figure 5-50 Exporting data to a fixed-length file

- 3 In Export target, select Fixed length file.
- 4 In Export, type the name of the fixed-length file.
- 5 In Resolution table, select the table from which you want to export data.
- 6 In Enter columns, type the name of the columns, using the required format. Leave it empty if you wish to export the whole table.
- 7 Choose Click to edit the domain to define a domain that would filter the data. Defining a domain is required.
- 8 In Description, type a short description of the instruction's purpose.
- 9 In Export, choose OK to define the Export instruction.
- 10 Save the project.
- 11 Run the Export instruction. The exported file is saved on the machine where BIRT Loader is installed. The file is located in:

C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium

The file format uses fixed length for saving the data column, as shown in Listing 5-2.

Listing 5-2 Fixed-length file

Melbourne	Victoria	Australia
Nantes		France
San Rafael	CA	USA
NYC	NY	USA
Madrid		Spain
Lyon		France

IFEXISTS

Conditionally executes an INCLUDE statement, depending on the value of an expression. The expression evaluates the existence of a database object, such as a database, a table, and a column. Depending on the result, it executes the true or false INCLUDE statement. The syntax is:

Syntax IFEXISTS [Object] [Include if true] [Include if false]

Parameters Object

A database object such as database, table, or column.

Include if true

The file name of a script file. If the database object exists, the instruction includes this script file in the load process.

Include if false

The file name of a script file. If the database object does not exist, the instruction includes this script file in the load process.

The example shown in Figure 5-51 checks if the [SalesData].[CarInventory] table exists. The Ifexists instruction executes the Export-file.xml script if the table exists, or Aggregation-scripts.xml, if the table does not exist.

Figure 5-51 Creating the Ifexists instruction

IFLINKED

Evaluates the existence of a link between two tables and executes an INCLUDE statement, depending on the result. This instruction allows you to dynamically execute scripts, depending on a condition. The syntax is:

Syntax `IFLINKED [From link] [To link] [Include if true] [Include if false]`

Parameters From link

Defines the name of the column from which the join is initiated. The column must be defined using the full name in the following format:

`[Database] . [Table] . [Column]`

To link

Defines the name of the column to which the link is created. The column must be defined using the full name in the following format:

`[Database] . [Table] . [Column]`

Include if true

The file name of a script file. If the link exists, the instruction includes this script file in the load process.

Include if false

The file name of a script file. If the link does not exist, the instruction includes this script file in the load process.

How to use IFLINKED

- 1 In Main file, choose New. In the list of transformation instructions, choose IFLINKED. If linked appears.
- 2 In From, define a database, a table, and a column from which the link initiates, as shown in Figure 5-52.
- 3 In To, define a database, a table, and a column, to which the link points.
- 4 In Include if true, select the script file to be executed if the link exists. Choose Refresh if you do not see your script in the list.
- 5 In Include if false, select the script file to be executed if the link does not exist. Choose Refresh if you do not see your script in the list.

The screenshot shows the 'If linked' dialog box. It has a title bar 'If linked'. Inside, there are two sections: 'Include if true' and 'Include if false'. Each section has a list box with two items: 'Aggregation-scripts.xml' and 'Export-file.xml'. The 'Export-file.xml' item is selected in both lists. Above the lists are 'From' and 'To' fields, each with three dropdown menus. The 'From' dropdowns are '[SalesData]', '[Customers]', and 'ustomerNumber'. The 'To' dropdowns are '[SalesData]', '[SaleOrders]', and 'ustomerNumber'. At the bottom, there is a 'Description' text box, a 'Cancel' button, and an 'OK' button.

Figure 5-52 Creating an Ilinked instruction

- 6 In If linked, choose OK. The Ilinked instruction appears in Main file.

INCLUDE

Includes a script file or creates an empty include entry.

Syntax `INCLUDE [File name]`

Parameter **File name**
The name of the script file you want to include in the main script.

How to include a file

The example includes a script file, as shown in Figure 5-53, to the main script.

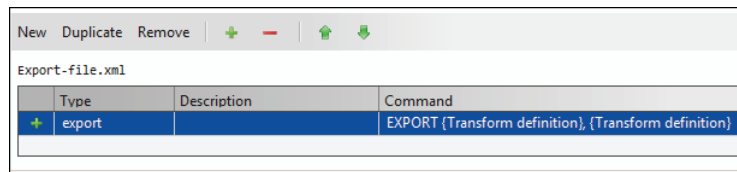


Figure 5-53 Creating a script file

- 1 In Main file, choose New. The list of transformation instructions appears.
- 2 In the list of transformation instructions, choose Include. Include appears, as shown in Figure 5-54.

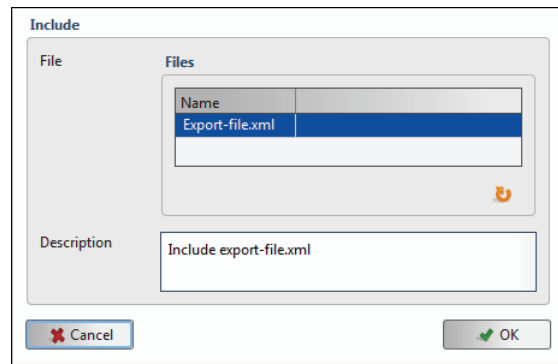


Figure 5-54 Including a script file

- 3 In Include, click Refresh to make the latest script files appear.
- 4 In Files, select the file you want to include.
- 5 In Description, write a short note explaining the instruction.
- 6 Choose OK. The instruction appears in Main file.
- 7 Save the project.

PRINT

Writes comments in a project log file. The syntax is:

Syntax PRINT [Text]

Parameter **Text**
The text to include in the log file.

For example, a comment, like the one shown in Figure 5-55, appears in the log files as:

03/05/2013 13:35:37-*INFO-Filter women under 30

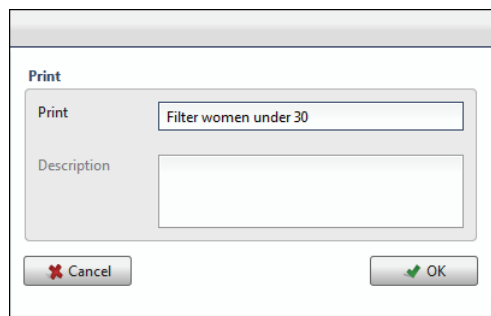


Figure 5-55 Creating a comment

VAR

Defines variables. Variables can be used to dynamically control script execution. You can store different types of values in variables. The syntax is:

Syntax VAR[Variable name] [Variable value]

Parameters **Variable name**

The name of the variable. Use descriptive names.

Variable value

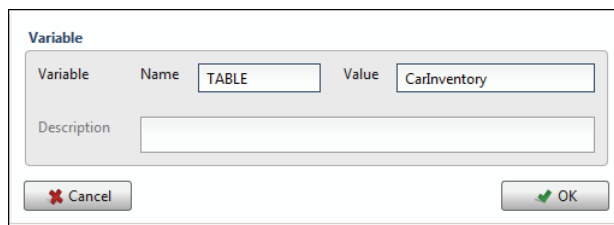
The value of the variable. The values can be of any type.

To reference a variable use the following syntax:

%VAR%

How to use a variable

- 1 In Main file, choose New. In the list of transformation instructions, choose VAR. Variable appears.
- 2 In Name, type the variable name, as shown in Figure 5-56. In Value, type the variable value. In this example, the variable contains the name of a table.



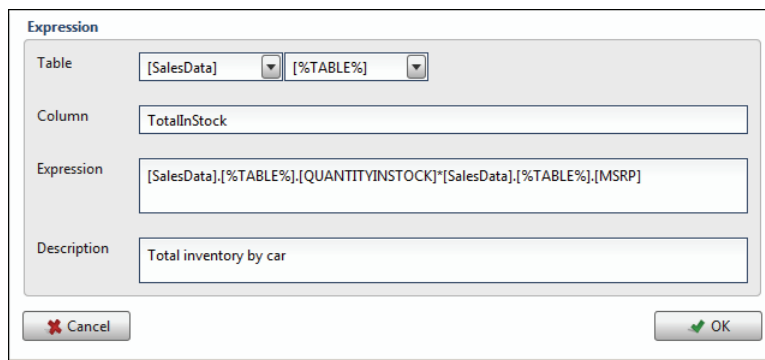
The dialog box titled 'Variable' contains the following elements:

- Name:** A text box containing the word 'TABLE'.
- Value:** A text box containing 'CarInventory'.
- Description:** An empty text box.
- Buttons:** 'Cancel' (with a red X icon) and 'OK' (with a green checkmark icon).

Figure 5-56 Creating a variable

- 3 In Variable, choose OK. Variable appears in Main file.
- 4 In Main file, choose New. In the list of transformation instructions, choose EXPRESSION. Expression appears.
- 5 In Table, select a database from the drop-down list, and type the following expression in the table box, as shown in Figure 5-57:

[%TABLE%]



The dialog box titled 'Expression' contains the following elements:

- Table:** A dropdown menu showing '[SalesData]' and a text box containing '[%TABLE%]'.
- Column:** A text box containing 'TotalInStock'.
- Expression:** A text box containing '[SalesData].[%TABLE%].[QUANTITYINSTOCK]*[SalesData].[%TABLE%].[MSRP]'.
- Description:** A text box containing 'Total inventory by car'.
- Buttons:** 'Cancel' (with a red X icon) and 'OK' (with a green checkmark icon).

Figure 5-57 Using a variable

- 6 In Column, type the name of the new column.
- 7 In Expression, type the expression for calculating the value of the new column. Use the variable reference where needed.

[SalesData] . [%TABLE%] . [QUANTITYINSTOCK] * [SalesData] . [%TABLE%] . [MSRP]

- 8 Choose OK, and select Save, to save the expression.

- 9 Choose Run. The log, as shown in Figure 5-58, displays that the TABLE variable is replaced by its value, CarInventory.

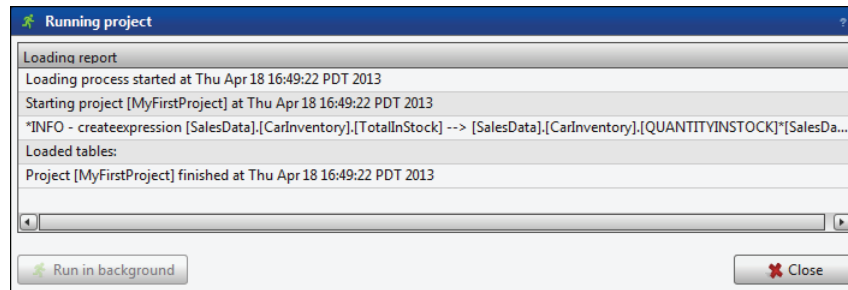


Figure 5-58 Running a project

Administering BIRT Loader

This chapter contains:

- About BIRT Loader administration
- Managing users
- Scheduling load processes
- Using External call
- Understanding the BIRT Loader internal database
- Copying a load project to a new location
- Managing the repository

About BIRT Loader administration

To access BIRT Loader Administration, you must have an administration account. Administration is located on the front page of the BIRT Loader, as shown in Figure 6-1, and appears only if the logged-in user is an administrator.



Figure 6-1 Starting Administration

The Administration page of BIRT Analytics Loader is shown in Figure 6-2. The main activities are presented as tabs:

- **Loader users**
Manages the BIRT Loader users.
- **Scheduled tasks**
Create and manage BIRT Loader scheduled tasks.
- **External call**
Returns a credential security file for a selected project. Using the file allows you to run a project remotely.

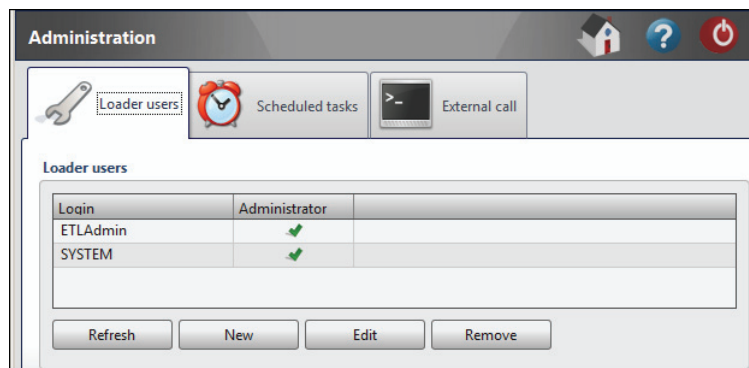


Figure 6-2 Administering BIRT Loader

Managing users

Loader users sets up user access to BIRT Analytics Loader. There are three options:

- **New**
Create a new BIRT Analytics Loader user, as shown in Figure 6-3.

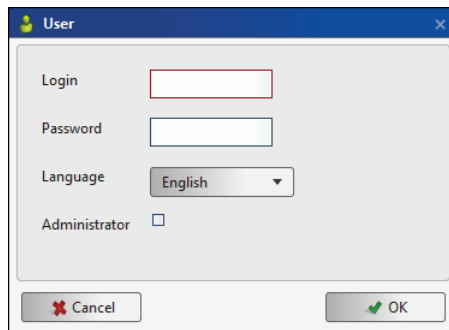

 A dialog box titled "User" with a close button (X) in the top right corner. It contains four fields: "Login" (a text input field), "Password" (a text input field), "Language" (a dropdown menu currently showing "English"), and "Administrator" (a checkbox). At the bottom, there are two buttons: "Cancel" with a red X icon and "OK" with a green checkmark icon.

Figure 6-3 Creating a user

For each user, you must define the following:

- Login
User name.
- Password
Password for the user.
- Language
The language in which you want the user interface to appear. Currently, only English is supported.
- Administrator
Indicate if the user is an administrator.
- Edit
Edit an existing BIRT Analytics Loader user.
- Remove
Remove a BIRT Analytics Loader user.

Scheduling load processes

As good practice, to not disturb the analysis process, load data when BIRT Analytics users are not in the system. Scheduling allows you to run the load processes overnight, or at times when the system is idle.

Scheduled tasks allows you to create and manage scheduled tasks, as shown in Figure 6-4.

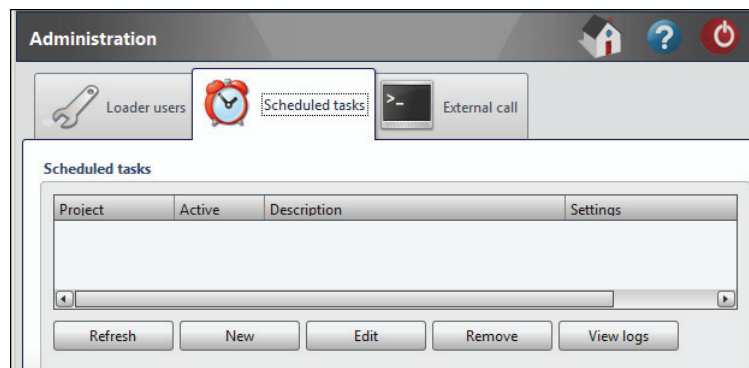

 A screenshot of the "Administration" window. At the top, there are three tabs: "Loader users" (with a wrench icon), "Scheduled tasks" (with a clock icon), and "External call" (with a terminal icon). The "Scheduled tasks" tab is selected. Below the tabs, there is a table with columns: "Project", "Active", "Description", and "Settings". The table is currently empty. Below the table, there are five buttons: "Refresh", "New", "Edit", "Remove", and "View logs".

Figure 6-4 Scheduling tasks

The following activities are available:

- Refresh
Refresh scheduled tasks.
- New
Create new tasks. When creating a new task, you must provide the following information, as shown in Figure 6-5:

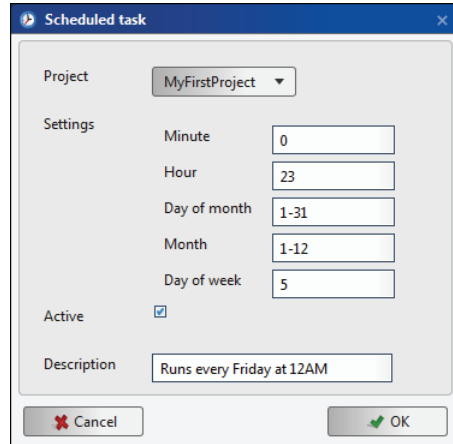


Figure 6-5 Creating a scheduled task

- Project
Select a project name.
- Settings
Time settings, such as minute, hour, day of month, month, and day of week. Time settings use the following syntax:
 - Minute
Minutes within the hour (0-59)
 - Hour
The hour of the day (0-23)
 - Day of month
The day of the month (1-31)
 - Month
The month (1-12)
 - Day of week
The day of the week (0-7), where both 0 and 7 are Sunday

To specify multiple values for one column, the following operators are available, as listed in Table 6-1, in the order of precedence.

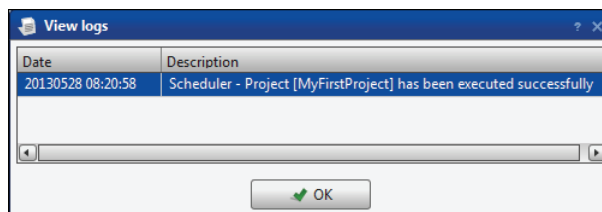
Table 6-1 Scheduling operators

Operators	Description	Examples
*	Match all values of a field.	Using an asterisk in Month indicates every month.
M-N	Define a range.	1-5 in Day of week means from Monday to Friday.

Table 6-1 Scheduling operators

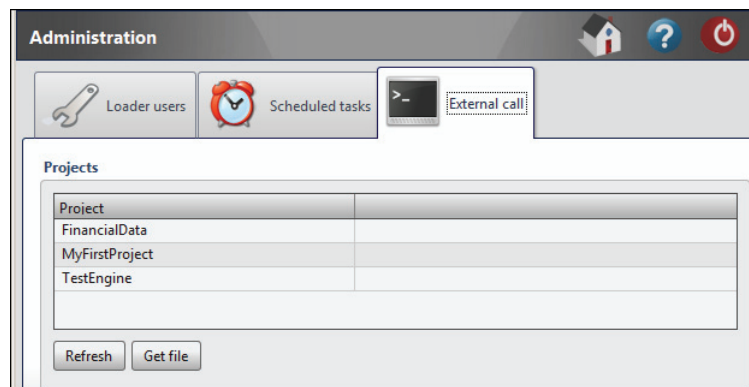
Operators	Description	Examples
*/X	Specify multiple values based on a start position and a period.	*/15 in Minute selects 0, 15, 30, 45 minutes. 3-59/15 in Minute indicates the third minute of the hour and every 15 minutes thereafter.
A,B,...,Z	Commas are used to separate items of a list.	0,30 for minutes, or 1,3,5 for days of the week.

- Active
The task will not run if Active is deselected.
- Description
A short note describing the scheduled task.
- Edit
Edit tasks.
- Remove
Remove tasks.
- View logs
View execution logs, as shown in Figure 6-6.

**Figure 6-6** Viewing scheduled task log

Using External call

External call returns a credential security file for a project, as shown in Figure 6-7. The credential file allows you to run a project remotely, from a command prompt, or a batch file. Typically projects contain load and transformation instructions. You can control what instructions are executed from Data and Transform options in Projects management.

**Figure 6-7** Getting a project credential file

The credential file contains encrypted information about the current date and time, the project name, and security information associated with the system where the product is installed. This information includes security keys that the BIRT Loader installer generates and saves in this folder:

```
C:\Program Files\Actuate11SP4\BIRTSAnalytics\dubnium\qloader
```

Public-key encryption uses two different mathematically related keys, known as a key pair, to encrypt and decrypt data. Information encrypted using one key can only be decrypted by using the other member of the key pair. The names of the keys are:

- `qloader.pbl`
Public key, associated with the system where the product is installed
- `qloader.prv`
Private key, used to decrypt the information encrypted with the public key

These files are only valid on the server where they are created. If these files are deleted, you can regenerate them by reloading the BIRT Loader web application to the Tomcat web server. Every time you reinstall the product, you must regenerate the load processes again.

The following activities are available:

- **Select project**
A drop-down list of the projects the user accesses.
- **Get file**
Generates a credential token for the selected project. This token is used to launch load processes remotely, or from batch files. By default, the project file name is `callProject.tkn`. You can change the name by choosing a name specific for a loading project.

Running projects using qLauncher

qLauncher is a Java-based utility, designed to run projects from batch files or a command prompt. The utility launches a load process remotely. The file, `qLauncher.jar`, is in the Tools folder of the installation files. `qLauncher.jar` is built using Java 1.7, and requires Java 1.7 in order to run.

The qLauncher command syntax is:

```
java -jar qlauncher.jar --url=[BIRT Loader Web Service URL]  
--file=[Credential project file name] --out=[log file name]
```

Parameters

--url

The BIRT Loader web service URL. This is a required parameter.

--file

The name of the project credential file. This is a required parameter.

--out

The name of the file to save the execution log. This is an optional parameter.

Example:

```
java -jar qlauncher.jar --url="http://localhost:8080/qloader/run"  
--file="callProject.tkn" --out="output.txt"
```

The `qLauncher.jar` file must be placed in the same folder as the project credential file. To obtain help for the command syntax, open a command prompt, and type the following:

```
java -jar qlauncher.jar
```

When you run a project, the utility changes the public key with the private key to log in to the BIRT Loader engine. To automate the data loading, include the command in a batch file.

How to run a project using qLauncher

This procedure runs a project remotely.

- 1 Generate a project credential file.
 - 1 Log in to Admin in BIRT Analytics Loader.
 - 2 In External call, select a project from the project list.
 - 3 Choose Get file. Opening callProject.tkn appears, as shown in Figure 6-8.

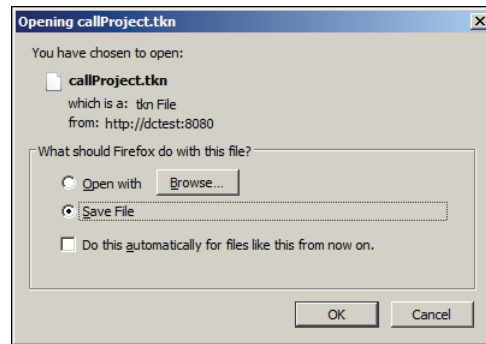


Figure 6-8 Opening callProject.tkn

In Opening callProject.tkn, choose OK. Depending on the web browser, the file is saved in the configured download folder.

- 2 Create a folder for the remote project, run on your computer, for example:

```
C:\Load\qLauncher
```

- 3 Copy qLauncher.jar from the Tools folder of your installation files to the loading folder.
- 4 Copy callProject.tkn to the loading folder.
- 5 Open a command prompt CMD.

- 6 In CMD, navigate to the project folder. For example, type:

```
cd C:\Load\qLauncher
```

- 7 In CMD, type:

```
java -jar qLauncher.jar
```

The help for the command syntax appears, as shown in Figure 6-9. This step is optional.

- 8 In CMD, type the command to run the project. In this example the command looks like this:

```
java -jar qlauncher.jar --url="http://dctest:8080/qloader/run"  
--file="callProject.tkn" --out="output.txt"
```

The project run finishes and the prompt appears, as shown in Figure 6-9.

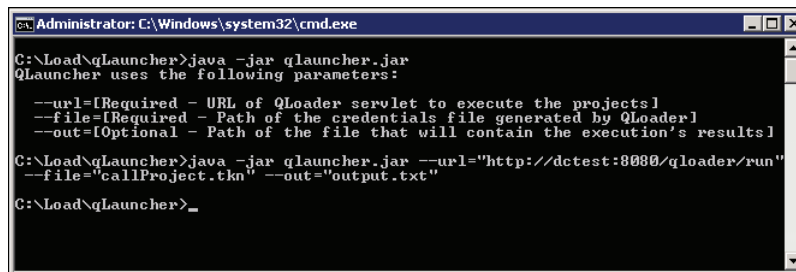


Figure 6-9 Running qLauncher.jar

- 9 qLauncher saves the log file, output.txt, in the current folder. Listing 6-1 shows the log outputs for the executed transactions.

Listing 6-1 output.txt

```
- Loading process started at Fri Apr 26 14:12:30 PDT 2013
04/26/2013 14:12:31 - Starting project [MyFirstProject] at Fri Apr 26
14:12:31 PDT 2013
04/26/2013 14:12:31 - *INFO - Table: [SalesData].[Customers]
04/26/2013 14:12:34 - *INFO - Table: [SalesData].[Customers] - 122
04/26/2013 14:12:34 - *INFO - Table: [SalesData].[CarInventory]
04/26/2013 14:12:35 - *INFO - Table: [SalesData].[CarInventory] - 110
04/26/2013 14:12:35 - *INFO - Table: [SalesData].[DailyPurchases]
04/26/2013 14:12:36 - *INFO - Table: [SalesData].[DailyPurchases] - 35
04/26/2013 14:12:36 - *INFO - Table: [SalesData].[MonthlySales]
04/26/2013 14:12:36 - *INFO - Table: [SalesData].[MonthlySales] - 109
04/26/2013 14:12:36 - *INFO - Table: [SalesData].[SaleOrders]
04/26/2013 14:12:38 - *INFO - Table: [SalesData].[SaleOrders] - 326
04/26/2013 14:12:46 - *INFO - Table: [SalesData].[Employees]
04/26/2013 14:12:47 - *INFO - Table: [SalesData].[Employees] - 14
04/26/2013 14:12:48 - *INFO - Table: [SalesData].[Clients]
04/26/2013 14:12:49 - *INFO - Table: [SalesData].[Clients] - 122
04/26/2013 14:12:49 - Loaded tables:
Customers,CarInventory,DailyPurchases,MonthlySales,SaleOrders,
Employees,Clients
04/26/2013 14:12:49 - Project [MyFirstProject] finished at Fri Apr 26
14:12:49 PDT 2013
```

Understanding the BIRT Loader internal database

BIRT Analytics Loader configuration information is stored in dubnium.db, located in:

C:\Program Files\Actuate\SP4\BIRTAnalytics\dubnium\qloader\dubnium.db

dubnium.db is a sqlite database. To access the sqlite database, you can use SQLite Manager, an add-in in the Firefox web browser. The main tables in dubnium.db are listed in Table 6-2.

Table 6-2 BIRT Loader internal database tables

Table	Description
PROJECTS	Load projects.
USERS	BIRT Analytics Loader users. By default, user SYSTEM is created.
USERS	Data sources connection properties.
FILES	Remote data provider definitions.
URIS	Remote data provider URIs.
URIPARAMS	Remote data provider URI parameters.
EXECUTIONS	BIRT Analytics Loader executions.
EXECUTIONLOG	Execution log details.
LOCKS	Users who locked BIRT Analytics Loader.
BACKUP	Backup of load projects.
SCHEDULER	Information about scheduled tasks.
SCHEDULERLOGS	Scheduled tasks logs.
VERSION	Database (dubnium.db) version.

The scripts, logs, and repository generated by BIRT Analytics Loader are stored in the FastDB installation folder. By default, these folders are:

- **Repository**

C:\Program Files\Actuate11SP4\BIRTSAnalytics\dubnium\databases\

- **Log Files**

C:\Program Files\Actuate11SP4\BIRTSAnalytics\dubnium\log\

- **Load Project**

C:\Program Files\Actuate11SP4\BIRTSAnalytics\dubnium\loading-projects\

Contains load projects. Every project is stored in a different folder. The names of these folders is a sequential number, such as 1, 2, 3, and so on.

A project folder contains the following sub-folders and files:

- **Data**

A folder where you upload and store source files with data.

- **Transformations**

A folder where the transform files are stored. These files are generated in XML format.

- **Definitions**

A folder where the definitions of files and column information are stored. These files are generated in XML format.

- **Descriptor.txt**

An XML file that contains the project name and its creation date.

- **Control.xml**

An XML file that contains the declaration of files to load.

Copying a load project to a new location

This section explains how to copy the files of a load project between two separate environments. Use this feature to move the project folder location, copy projects between machines, and move projects from development to production.

The project folders are named using sequential numbers like 1, 2, 3, and so on. You must identify your project by its content.

How to copy a load project

1 Stop Dubnium Server.

1 In Windows→Start, type:

services.msc

Services appears, as shown in Figure 6-10.

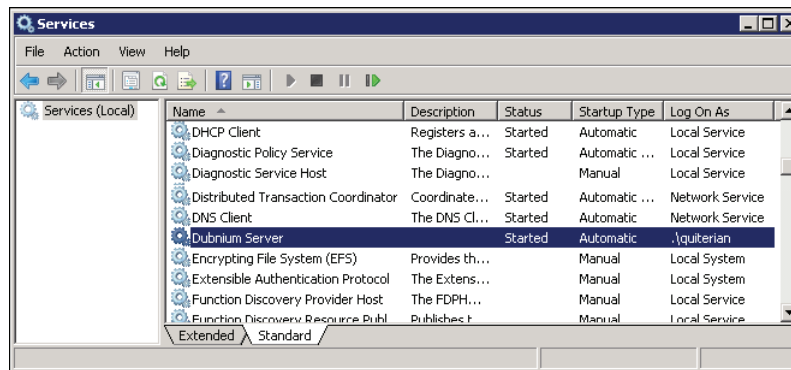


Figure 6-10 Stopping Dubnium Server

- 2 In Services, select Dubnium Server, and choose Stop from the right-click menu.
- 2 Create a new empty load project using BIRT Analytics Loader. The configuration.xml, as shown in Listing 6-2, defines the path where the load project is stored.

Listing 6-2 Configuration.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <projectspath>C:\\Program Files\\Actuate11SP4\\BIRTAanalytics\\dubnium\\
    \\loading-projects</projectspath>
  <descriptor>descriptor.txt</descriptor>
  <urlclient>fastdbloader/build/client.jsp</urlclient>
  <urladmin>admin/build/client.jsp</urladmin>
  <title>Loader</title>
  <transformationslist>transformations.xml</transformationslist>
  <ignoresslerrors>1</ignoresslerrors>
  <!-- uploading files. Unit is Kb -->
  <maxmemsize>1024</maxmemsize>
  <maxfilesize>51200</maxfilesize>
  <connectionstringsfile>connectionstrings.xml</connectionstringsfile>
</configuration>
```

- 3 Copy your old project files, except descriptor.txt file, to the new blank project location.
- 4 Start Dubnium Server.
- 5 Open the load project from BIRT Analytics Loader and check that it works properly.

Managing the repository

The FastDB database installation creates a default data repository in this path:

C:\Program Files\Actuate11SP4\BIRTAanalytics\dubnium\databases\

You have the option to work in the default repository or create a new one. To back up a repository, copy the content of the \databases folder to the backup location. In case of a large repository, compress the repository folder to minimise the required space. Larger repositories require more time to compress and copy. Consider this factor when you plan a backup because your system will not be accessible during the operation.

How to back up a repository

- 1 Stop Dubnium Server.
- 2 Make a copy of the current repository content in the following location:

C:\Program Files\Actuate11SP4\BIRTAanalytics\dubnium\databases

3 Start Dubnium Server.

Creating a new database repository

To create a new FastDB database you must specify the database location in the configuration files. At installation, the default database contains the Demo database, and is located at:

C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium\databases

How to create a new repository

1 Stop Dubnium Server.

1 In Windows→Start, type:

services.msc

Services appears, as shown in Figure 6-10.

2 In Services, select Dubnium Server, and choose Stop from the right-click menu.

2 In Windows Explorer, create a new folder where you want to store the Dubnium database. For example:

C:\Production\SalesData

3 Set privileges for the user running the Dubnium server to this folder.

1 In Windows Explorer, select the folder, and choose Properties from the right-click menu.

2 In Properties, choose Security.

3 In Security, choose Edit.

4 In Group or user names, select the user running the Dubnium server, as shown in Figure 6-11. If the user is not in the list, choose Add and select the user to be added.

5 In Permissions for <user>, in Full control, select Allow. Choose Apply. Choose OK.

In Security, choose OK.

In Properties, choose OK.

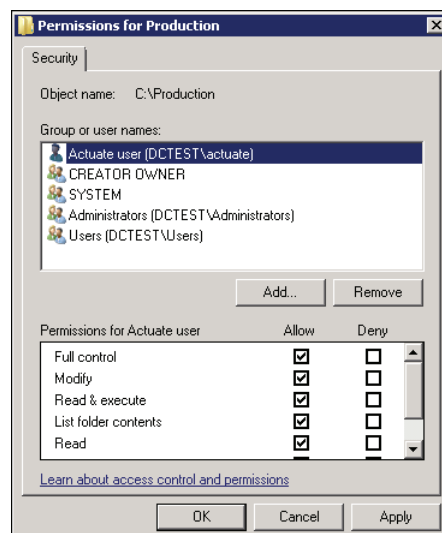


Figure 6-11 Setting folder privileges

4 Open Windows Explorer, and navigate to:

C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium

- 5 Open the engine_configuration.ini file in Notepad. The file looks like the one in Listing 6-3.

Listing 6-3 engine_configuration.ini

```
[engine]
repository = C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnum
\repositories
[log]
path = C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnum\log\engine.log
verbosity = 6
[server]
admin_file_path = C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnum
\admin.sqlite
[memory]
maxmemorysystem = 2048
maxmemorythread = 1024
```

- 6 Modify the following line:

```
repository = C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnum
\repositories
```

to:

```
repository = C:\Production\SalesData
```

- 7 Save and close the file.
- 8 In Services, select Dubnum Server and choose Start from the right-click menu.

Copying a data repository to a new location

BIRT Analytics supports changing the location of a data repository by copying the repository to a new location. The Dubnum server must be shut down before copying the repository. Before restarting the Dubnum server, you must change the FastDB engine configuration file to specify the new location.

How to copy a data repository to a new location

- 1 Stop Dubnum Server.
- 2 Open Windows Explorer, and navigate to:
C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnum
- 3 Open the engine_configuration.ini file in Notepad.
- 4 Modify the following line:
repository = C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnum
\repositories
for example, to:
repository = C:\Production\SalesData
- 5 Set privileges for the user running the Dubnum server to this folder, in this case:
C:\Production\SalesData
To set privileges, follow the procedure in step 3 in “How to create a new repository,” earlier in this chapter.
- 6 Save and close the file.
- 7 Copy all files and folders from the old repository, which by default is:
C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnum\repositories\

to the newly defined repository:

```
C:\Production\SalesData\
```

- 8 Start Dubnium Server.
- 9 Synchronize and assign permissions using BIRT Analytics Administration, as described in “Synchronizing data” in Chapter 2, “Working with projects.”
- 10 Open BIRT Analytics and check that it works properly.

Changing the repository content using the same location

You can change the repository content without changing the location. For example, you have a backup of a BIRT Analytics repository, and you need to restore the data to the same repository location. In this case, it is not necessary to change the configuration file, you only change the content of the repository folder.

How to change the content of a repository

This procedure restores the content of the FinancialData repository to the current Demo repository. The FinancialData backup is in the C:\Financial folder.

- 1 Stop Dubnium Server.
- 2 Make a copy of the current repository content in:

```
C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium\databases
```
- 3 Delete the current repository files in:

```
C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium\databases
```
- 4 Copy the content of:

```
C:\Financial
```


to:

```
C:\Program Files\Actuate11SP4\BIRTAnalytics\dubnium\databases
```
- 5 Start Dubnium Server.
- 6 Synchronize and assign permissions using BIRT Analytics Administration, as described in “Synchronizing data” in Chapter 2, “Working with projects.”
- 7 Log in to BIRT Analytics and BIRT Loader to verify the restore. Refresh the repository view, if needed.

Adding a BIRT Analytics database to a repository

BIRT Analytics supports copying databases from one repository to another. To do this you must first export the database to a text delimiter file format. The export procedure is described in “EXPORT” in Chapter 5, “Transforming data.” Then, import the files to a new database in a new repository using the procedures described in Chapter 3, “Extracting data.” Another option for importing data is to use the Import feature in BIRT Analytics. For more details about this option, see *Using BIRT Analytics*.

A

access control list (ACL)

A group or set of users with access to a database object. Using the BIRT Analytics Administration tool, the administrator creates a security group or ACL that manages privileges for a database object.

Related terms

BIRT Analytics Administration, column, database, group, security role, table

action

An action is an event executed by a manual or task trigger. Example actions include send e-mail, query action, delete column, and apply model.

Related terms

scheduled task, trigger

Aggregates

A tool that supports grouping data from multiple tables in one table. Aggregates supports defining a function and filter as properties.

Related terms

filter, table

analysis

A tool that provides a specific view of data stored in FastDB. BIRT Analytics supports multiple analyses.

Related terms

Bubble analysis, Crosstab analysis, Evolution analysis, FastDB, Map analysis, Pareto analysis, Profile analysis, Venn analysis

antecedent

Terms representing the left-hand, or If... clause of an association rule. The antecedent clause of an association rule contains discrete data items.

Related terms

association rules, consequent

association rules

A predictive analytics technique that analyzes data for frequent If...Then patterns and calculates support and confidence criteria that identify the most important relationships. Support indicates how frequently the items appear in the database. Confidence indicates the number of times the If...Then relationships evaluate true.

An association rule has two parts, an antecedent and a consequent. The antecedent represents one or multiple data items. The consequent represents an item found in combination with the antecedent. An association rule returns a lift and a leverage value that measure how well the rule predicts the consequent.

Related terms

antecedent, Association Rules, confidence, consequent, lift, leverage, predictive analytics, support

Association Rules

A predictive analytics tool that uses association rules to identify an If...Then relationship between data values stored in an information repository. For example, an association rule may show the following relationship: If a customer buys products A and B, then the customer also buys product C.

Related terms

association rules, predictive analytics

B

baseline filter

A filter that returns a group of records to serve as a basis for comparison. For example, use the year 2012 as a baseline filter for profit, to compare profit earned in another year with profit earned in 2012.

Related terms

filter, record

big data analysis

The practice of analyzing, exploring, filtering, loading, segmenting, and studying massive quantities of data. Big data analysis uses statistics to describe qualities and predict trends in these data repositories.

Related terms

analysis, BIRT Analytics, data repository

BIRT Analytics

An application, including a data repository, data loader, and web service, that supports big data analysis.

Related terms

big data analysis, BIRT Analytics Administration

BIRT Analytics Administration

A BIRT Analytics system administration tool that runs as a browser-based application. The administrative user has full permission to modify all configurable features of the BIRT Analytics system.

Related term

BIRT Analytics

BIRT Analytics Loader module

A tool that extracts, transforms, and loads records from an external data source to FastDB.

Related terms

BIRT Analytics, BIRT Analytics Administration, FastDB

Bubble analysis

A tool that supports viewing a spatial distribution of data with respect to two axes.

Related terms

analysis, Crosstab analysis, Evolution analysis, Map analysis, Pareto analysis, Profile analysis, Venn analysis

C

calculated field

A data field that displays the result of an expression.

campaign	<p>A set of tasks, defined for specific population segment. A campaign is completed during a defined time period to accomplish a specific goal.</p> <p>Related term segment</p>
Canvas	<p>A workspace for data analysis gadgets. Canvas supports arranging, assembling, and saving a collection of data visualization gadgets.</p> <p>Related term gadget</p>
cell	<p>A set of properties that defines campaign actions to be performed for all records in a segment.</p> <p>Related terms action, campaign, record, segment</p>
Clustering	<p>A predictive analytics tool that uses k-means cluster analysis. Clustering identifies groups of similar data values in large segments stored in a big data repository.</p> <p>Related terms k-means, cluster analysis, predictive analytics</p>
cluster analysis	<p>A data analysis task that iterates estimating of values assigned to common data attributes. Common attributes identify groups of similar items, called clusters. Comparing clusters highlights similar and different groups in big data.</p> <p>Related terms analysis, big data analysis, Clustering</p>
column	<p>1 A named field in a database table or query. For each data row, the column can have a different value, called the column value. The term column refers to the definition of the column, not to any particular value.</p> <p>2 A vertical sequence of cells in a crosstab, grid element, or table element.</p> <p>Related terms column-oriented DBMS, database, data field, query, table</p>
column-oriented DBMS	<p>A column-oriented DBMS is a database management system (DBMS) that stores data tables as sections of columns of data rather than as rows of data. A column-oriented DBMS serializes all of the values of a column together, then the values of the next column, and so on.</p> <p>Related terms database, column</p>
confidence	<p>An expression used to identify an association rule. Confidence compares how often the consequent appears when the antecedent is met. The confidence expression has the following syntax:</p> $\text{Confidence } (A, B \rightarrow C) = \text{Support } (A, B, C) / \text{Support } (A, B)$ <p>Related terms association rules, support</p>
consequent	<p>Terms representing the left-hand, or ...Then clause of an association rule. The consequent clause of an association rule contains items found in combination with items in the antecedent.</p> <p>Related terms antecedent, association rules</p>
Convert	<p>A BIRT Analytics option that displays results from one data analysis using a different type of data analysis. For example, an analysis created using Crosstab converts to a Bubble, Evolution, or Map analysis.</p>

Related term
analysis

count The total number of records in a field.

Related terms
field, record

Crosstab analysis

A tool that supports analyzing data using cross-tabulation, or pivoting of different fields.

Related terms
analysis, Bubble analysis, Evolution analysis, Map analysis, Pareto analysis, Profile analysis, Venn analysis

Cylinder A data visualization gadget that displays numeric values and boundaries in ranges. A Cylinder displays defined data measures as colored slices that comprise one cylinder shape.

Related terms
Dial, Funnel, gadget, Gallery, Label, Meter, Sphere

D

data analysis A process including acquiring, organizing, transforming, and modeling data to support decision-making.








Data Explorer

A tool that displays records from a database stored in FastDB. Data Explorer provides a summary view for a table and a detail view for records, tables, selections, and segments.

Related terms
Data Tree, FastDB, record, table

data field A location storing data having a specific type. A data field typically contains data from a database or other data source. A data field appears as a column when viewing a table in Data Explorer. For example, the BIRT Data Analytics Demo database includes the data field types listed in Table G-1.

Table G-1 Data field types

Icon	Field type	Description
	Calculated	Displays a value result from an expression
	Date	Contains numbers that represent day, month, and year
	Date and time	Contains numbers that represent day, month, year, and time of day
	Full numeric	Contains whole, or integer numbers, such as 1 or 1000
	Real numeric	Contains real, or partial numbers such as 1.05 or 0.003
	Time	Contains a value representing time of day
	Text	Contains a string of alphabetic characters

Related terms
record, Data Explorer, Data Tree, data types

data integration

A process through which data in varied sources is combined.

data mining

A computational process used to extract and transform data to prepare it for analysis.

Related term

analysis

data repository

A physical or virtual location for storage and retrieval of data.

Related term

FastDB

Data Tree

A tool that supports viewing and working with databases, tables, and records stored in FastDB. Data Tree includes Discrete Values, My Data, and My Folders viewers.

Related terms

database, Discrete Values Viewer, My Data Viewer, My Folders Viewer, record, table

data types

A data type defines the limits of a data field in a BIRT Analytics database. For example, the BIRT Data Analytics demo database includes the data types listed in Table G-2.

Table G-2 Data types in BIRT Analytics Loader

Data type	Description
Date	Contains numbers that represent day, month, and year. The default format is mm_dd_yyyy.
Datetime	Date and time data from January 1, 1753, through December 31, 9999, providing accuracy to three-hundredths of a second, or 3.33 milliseconds. The default format is yyyy_mm_dd_hh_MM_ss.
Integer	Integer data from -2^{31} (-2,147,483,648) through $2^{31}-1$ (2,147,483,647).
Longint	Integer data from -2^{63} (-9,223,372,036,854,775,808) through $2^{63}-1$ (9,223,372,036,854,775,807).
Real	Floating precision number data with the following valid values: $-3.40\text{E}+38$ through $-1.18\text{E}-38$, 0 and $1.18\text{E}-38$ through $3.40\text{E}+38$.
String	A sequence of ASCII characters.
Time	Contains a value representing time of day. The default format is hh_MM_ss.
Unicode	A sequence of characters based on consistent encoding, representation, and handling of text as expressed in global writing systems.

Related terms

Data Explorer, data field, Data Tree, record

database

1 An integrated collection of logically related records that provides data for information application platforms, such as BIRT. The database model most commonly used is the relational model. Other typical models are entity-relationship, hierarchical, network, object, and object-relational.

2 An integrated set of logically related records stored in FastDB.

Related terms

record, table

decision tree

A predictive analytics technique that predicts the value of a target variable, based on values of multiple input variables. For example, use a decision tree to predict a survival rate, based on characteristics of the population that may survive.

Related terms

Decision Tree, predictive analytics

Decision Tree

A predictive analytics tool that uses the decision tree technique to predict an outcome, based on values of multiple input variables. For example, use Decision Tree to predict the product a customer will purchase, based on customer, purchase, gender, occupation, and income data.

Related terms

association rules, predictive analytics

Decodes

A tool that supports renaming a data field stored in FastDB.

Related terms

data analysis, data field, FastDB

Dial

A data visualization gadget that uses a needle-shaped pointer to display defined measures and numeric values in a range.

Related terms

Canvas, Cylinder, Funnel, gadget, Gallery, Label, Meter, Sphere

Discrete Values Viewer

A tool that supports viewing discrete values in a data record, selection, or segment.

Related terms

My Data Viewer, My Folders Viewer, record, segment, selection

Downloads

A tool that supports writing FastDB records to an external database.

Related terms

database, Export file, FastDB, record

Dubnium.exe

The file that runs the BIRT Analytics data repository, FastDB.

Related term

FastDB

E

Evolution analysis

A tool that supports viewing a time-progression view of data values.

Related terms

analysis, Bubble analysis, Crosstab analysis, Map analysis, Pareto analysis, Profile analysis, Venn analysis

Export Analytic DB

A tool that supports creating a new database field based on a segment defined in the database. The new field is stored in FastDB.

Related terms

Export file, FastDB, segment

Export file

A tool that supports creating a new text file based on a segment defined in the database. The file is stored in FastDB.

Related terms

Downloads, FastDB, segment

Expressions

A tool that supports creating a logical relationship, using data fields, functions, and operators. Results of the relationship appear as a calculated field in FastDB.

Related terms

calculated field, data field, FastDB

F

FastDB	The BIRT Analytics data repository. FastDB is a web service that caches data and supports executing data analysis and forecasting algorithms. Related terms Data Tree, database, data repository, record, table
field	See data field.
filter	A function that limits the number of records included a segment or selection. BIRT Analytics supports the following three filter types: baseline, target, and universal. Related terms baseline filter, target filter, universal filter
Forecasting	A predictive analytics tool that uses the Holt-Winters, iterative method. Forecasting predicts a future trend in data exhibiting a seasonal pattern. Related terms Holt-Winters, predictive analytics
functionalities	The system privileges an administrator grants to a security role. Related terms BIRT Analytics Administration, security role
Funnel	A data visualization gadget displaying numeric values and boundaries that represent groups in a range, using colored bands that display on a funnel shape. Related terms Cylinder, Dial, gadget, Gallery, Label, Meter, Sphere

G

gadget	A computer program that provides services without requiring an application for each one. BIRT Analytics provides multiple gadgets that support data visualization. Related terms Cylinder, Dial, Funnel, Gallery, Label, Meter, Sphere
Gallery	A tool that supports running multiple data visualization gadgets. Use the Gallery to assemble, arrange, and save gadgets on the Canvas. Related terms Canvas, Cylinder, Dial, Funnel, gadget, Label, Meter, Sphere
group	A set of users belonging to the same organizational unit who share the same permissions for performing tasks. Using the BIRT Analytics Administration tool, the administrator creates a group from the list of available users on the system. Related term BIRT Analytics Administration

H

has seasonality	User-selected option that recognizes a seasonal trend in a data set. Related terms Holt-Winters, seasonal periodicity, seasonality
Holt-Winters	A popular numerical estimation method used to forecast values in data that exhibit seasonal trends. The Holt-Winters method repeats and refines a time-series formula that includes a

level, trend, and seasonal component. The formula calculates forecast values valid for time t using a weighted average for all data prior to time t .

Related term

Forecasting

I

Import A tool that supports adding a field to a database by uploading records from an external database. The field is stored in FastDB.

Related terms

database, FastDB, field

indexed field A data field having an associated key. An indexed field appears in a summary table used for data retrieval.

Related terms

data field, field, table

J-K

k-means An iterative method of cluster analysis that groups large data sets into clusters of similar data. A k-means method forms clusters around data values having the nearest mean.

Related terms

analysis, Clustering, cluster analysis, mean

kurtosis A coefficient that describes the degree of concentration for a distribution of values, based on a mathematical average. The kurtosis coefficient is a value between -0.5 and 0.5. Colloquially, the Kurtosis coefficient is an average that indicates how sharp a distribution is with respect to a standard normal distribution.

Related terms

skewness, standard normal distribution

L

Label A data visualization gadget that associates specific alphanumeric characters with a defined measure. A Label displays a text description of a measure in the BIRT Analytics Gallery.

Related terms

Canvas, Cylinder, Dial, Funnel, gadget, Meter, Sphere

leverage A value that indicates how well an association rule predicts the consequent. The method used to calculate leverage differs from the method used to calculate lift.

Related terms

association rules, lift

lift A value that indicates how well an association rule predicts the consequent. A lift value greater than one indicates that the items in the rule appear together more than expected. The method used to calculate lift differs from the method used to calculate leverage.

Related terms

association rules, leverage

Links A tool that supports maintaining links binding columns and tables in a database stored in FastDB.

Related terms

column, table

M

make permanent

A field operation that creates a new data field from either a calculated field or a current segment. The data field appears in FastDB.

Related terms

calculated field, FastDB, field, segment

Map analysis

A tool that supports plotting data values and regions on a geographic map. For example, a map analysis shows geographic regions and the number of high-net-worth customers in each region.

Related terms

analysis, Bubble analysis, Crosstab analysis, Evolution analysis, Pareto analysis, Profile analysis, Venn analysis

maximum

The highest registered value in a set of values.

Related term

minimum

mean

An arithmetic mean of all registered values in the field.

Related terms

median, mode

median

A value that divides a field into two symmetrical parts.

Related terms

mean, mode

Meter

A data visualization gadget that uses colored bars to display numeric values and boundaries in a range.

Related terms

Canvas, Cylinder, Dial, Funnel, Gallery, Label, Sphere

minimum

The lowest registered value in a set of values.

Related term

maximum

mode

The values having the most frequent number of occurrences in a field.

Related terms

mean, median

My Data Viewer

A tool that supports viewing fields and tables in multiple databases stored in FastDB.

Related terms

Data Explorer, database, Discrete Values Viewer, field, My Folders Viewer, table

My Folders Viewer

A tool that supports viewing reports, selections, and gadgets by a user or, if shared, by other users. My Folders appears as a tab in Data Tree and in the Start pane.

Related terms

Data Explorer, Discrete Values Viewer, gadget, My Data Viewer, selection

N

NetScaler Web Logging (NSWL) query

A type of SQL query that tracks HTTP data traffic and writes information to a log file in a standard format such as the following example:

```
Select * from [Demo].[Household]
where [Demo].[Household].[Town]='LONDON';
```

Related terms

BIRT Analytics Administration, query, security filter, SQL (Structured Query Language)

normal distribution

A bell-shaped, single-peaked, symmetric distribution of data. In a normal distribution, the mean, mode, and median coincide at the center.

Related term

standard normal distribution

Numeric Ranges

A tool that supports creating a calculated field that includes a series of ranges into which data from numeric fields is grouped. For example, Numeric Ranges supports defining the following age ranges: Young - for age values less than 21, Adult - for age values 21 through 67, and Old - for age values greater than 67.

Related terms

calculated field, field

O-P

parameter

A variable expression that accepts a defined set of values.

Related term

filter

Parametric

A tool that supports creating a field based on a defined condition, for use as a filter on a measure.

Related term

field

Pareto analysis

A tool that supports comparing data using the Pareto principle, a commonly accepted rule which implies a data distribution with a numeric ratio of 80% to 20%. For example, the Pareto principle implies that 80% of sales result from 20% of customers.

Related terms

analysis, Bubble analysis, Crosstab analysis, Evolution analysis, Map analysis, Profile analysis, Venn analysis

predictive analytics

A subject encompassing a variety of techniques used to analyze current and historical facts to make predictions about future, or otherwise unknown events. Credit scoring is a well-known application that uses predictive analytics techniques to generate a score for an individual, based on credit history data for that individual.

Related terms

Association Rules, Clustering, Decision Tree, Forecasting

Profile analysis

A tool that supports identifying a set of similar characteristics in a group. A profile analysis compares z-score values calculated for each set of characteristics.

Related terms

analysis, Bubble analysis, Crosstab analysis, Evolution analysis, Map analysis, Pareto analysis, Venn analysis, z-score

profile

A set of associated security roles, groups, filters, and users. Using the BIRT Analytics Administration tool, the administrator creates a profile from the lists of roles, groups, filters, and users available on the system. From the BIRT Analytics security options list, choose Profiles, specify a profile name, provide a description, then select the roles, groups, filters, and users to include in the profile.

Related terms

BIRT Analytics Administration, group, security filter, security role

prompted filter

A data set filter that supports user entry of parameter values.

Related terms

filter, parameter

Q

Quantile

A tool that supports creating a new calculated field by grouping values in a numeric field, using multiple groups that contain an equal number of values. For example, use Quantile to group a field containing 2400 values into four quartiles having 600 values each.

Related terms

calculated field, field

query

A statement specifying the data rows to retrieve from a data source. For example, a query that retrieves data from a database typically is a SQL SELECT statement.

Related terms

database, SQL (Structured Query Language)

R

Ranking

A tool that supports ordering a table by generating a column of calculated values that correspond to a sorted column. The calculated values represent an ordered list of ranks.

Related terms

column, table

record

A set of related, indexed data fields in a database. A record often appears as a row shown in a table. For example, a customer record could include a numeric field for customerID, a character string field for customer name, and an alphanumeric field for age group.

Related terms

field, row

row

See record.

Related terms

field, record

S

scheduled task

A scheduled task includes a trigger, task details, and an assigned action.

Related terms

action, campaign, stage, trigger, workflow

Scratchpad	<p>A BIRT Analytics work area that supports temporary caching of multiple segments. Scratchpad also supports creating new fields based on segments or selections.</p> <p>Related terms Data Explorer, Data Tree, segment, selection</p>
seasonal periodicity	<p>A value indicating the number of periods in a cycle. Input a value for seasonal periodicity to initiate a forecast that predicts a seasonal pattern in a data set.</p> <p>Related terms Forecasting, Holt-Winters</p>
seasonality	<p>In a data set, a periodic trend that corresponds to monthly, quarterly, or semi-annual periods such as seasons.</p> <p>Related terms Forecasting, Holt-Winters</p>
security role	<p>A set of functionalities that an administrator uses to configure permissions in the BIRT Analytics system.</p> <p>Related terms BIRT Analytics Administration, functionalities, query, security filter</p>
security filter	<p>A type of query that an administrator uses to limit access to data in the BIRT Analytics system.</p> <p>Related terms BIRT Analytics, BIRT Analytics Administration, group, NetScaler Web Logging (NSWL) query</p>
segment	<p>A segment is a group of records sharing at least one common characteristic.</p> <p>Related terms record, selection</p>
selection	<p>A selection is a user-specified request that returns a segment from a database.</p> <p>Related terms record, segment</p>
skewness	<p>A value that reflects the distribution of values in a data set. Skewness values can be positive, zero, or negative. A positive value reflects a data set in which more values lie to the left of the mean value. A negative value reflects a data set in which more values lie to the right of the mean. A zero value indicates values distributed evenly around the mean, typically implying a symmetric distribution.</p> <p>Related terms kurtosis, mean</p>
Sphere	<p>A data visualization gadget that uses a colored sphere shape to display numeric values and boundaries in a range.</p> <p>Related terms Canvas, Cylinder, Dial, Funnel, gadget, Label, Meter</p>
SQL (Structured Query Language)	<p>A language used to access and process data in a relational database.</p> <p>Related term database</p>
stage	<p>A tool that supports defining users as task owners and assigning to each task owner the permissions required to perform tasks. Define a stage to identify part of a campaign.</p> <p>Related terms action, campaign, scheduled task, trigger, workflow</p>

standard deviation

The value equal to the positive square root of variance calculated for a data set.

Related term
variance

standard normal distribution

The normal distribution in which the mean is zero and the standard deviation is one.

Related term
normal distribution

Standardize column

A tool for preprocessing data values having a distribution different from a standard normal distribution. Multiple options support value sets distributed closely, clustered, spread, or having many repeated values.

Related term
standard normal distribution

sum

The cumulated sum of all the values in a field.

Related term
sum-of-squares

sum-of-squares

The sum of all of the squared values in a set.

Related term
sum

support

An expression that calculates a ratio measuring how many transactions contain all items in an association rule. The support expression has the following syntax:

$\text{Support (A,B)} = \text{Transactions (A,B)} / \text{Total transactions}$

Related term
association rules

T**table**

A named set of records in a database.

Related terms
database, record

target filter

A filter that returns a group of records for comparison with an established baseline. For example, use the year 2010 as a target filter for profit, to compare profit earned in 2010 with profit earned in another, baseline year.

Related terms
baseline filter, universal filter

temporal file

A temporary data file generated and stored in the system cache. Using the BIRT Analytics Administration tool, the administrator can remove the accumulated temporal files and records created by an application to optimize performance.

Related term
BIRT Analytics Administration

trigger

A trigger is a time or event that starts a scheduled task.

Related terms
action, campaign, scheduled task, stage, workflow

U

universal filter

A filter that is always applied at a lower resolution level, before changing resolution.

Related terms

baseline filter, target filter

V-Y

value

1 The content of a constant, parameter, symbol, or variable.

2 A specific occurrence of an attribute. For example, blue is a possible value for an attribute color.

Related term

parameter

variance

A value equal to the squared average of the distances between each value and the arithmetic mean.

Related term

mean

Venn analysis

A tool that supports data analysis based on crossing more than two fields. A Venn analysis identifies coincident values in multiple data segments. For example, use a Venn analysis to show how many customers buy the same three products.

Related terms

Bubble analysis, Crosstab analysis, Evolution analysis, Map analysis, Pareto analysis, Profile analysis, segment

W

workflow

A role responsible for completing tasks or stages in a campaign.

Related terms

action, campaign, scheduled task, stage, trigger

Z

z-score

A value describing whether a quantifiable difference between two groups is statistically significant.

Related term

Profile analysis

Index

Symbol

, (comma) character 91
* operator 60, 90
/ (slash) character 91
/ operator 60
+ operator 60
- (hyphen) character 90
- operator 60

A

ABS function 61
access control list 101
access control lists 13
access permissions. *See* privileges
access restrictions 10, 18
accessing
 BIRT Analytics Admin application 13, 88
 BIRT Analytics Loader 3
 data 20
 Dubnium database 94
 projects 8
 text files 25
 Transform text editor 53
accounts 4, 10, 18, 89
ACL to objects option 15
ACL to objects page 15
ACLs. *See* access control lists
actions 31, 101
Active property 91
Actuate BIRT Analytics 2
 See also BIRT Analytics
adding
 aggregate columns 53, 54
 calculated columns 60
 calculated fields 111
 data fields 110
 data sources 21
 database tables 31
 decoding columns 55
 domains 57, 58
 links 47, 75
 load processes 28–29
 parametric columns 67
 quantile range columns 69, 70
 ranking columns 71, 72
 remote data providers 25
 user accounts 18
 users 10, 88
Admin application. *See* BIRT Analytics
 Administration
Admin module (Loader) 18
Admin page (Analytics Admin) 14
Admin tab (Loader) 10

Admin tools 102
Administration page (Loader) 88
Administration tool. *See* BIRT Analytics
 Administration
administrator accounts 4
Administrator property 89
Administrator tab (Loader) 4
administrators 3, 6, 88
AGE function 61
Aggregate dialog box 54
aggregate filters 54
AGGREGATE instruction 53
Aggregates tool 101
aggregation functions 53, 54
ALLMONTHS function 61
ALLQUARTERS function 61
analysis (defined) 101
analysis gadgets. *See* gadgets
analytical functions 53
Analytics Admin tool. *See* BIRT Analytics
 Administration
Analytics application. *See* BIRT Analytics
Analytics Loader. *See* BIRT Analytics Loader
analytics processes 2
 See also ETL processes
analyzing data 2, 10
AND operator 61
antecedent 101
application servers 2
Association Rules 102
association rules 101
Attributes tab (Explorer) 43
authentication 17, 18
authentication files 17
automating
 load processes 92
 synchronization processes 17
Average function 54

B

backing up FastDB database 96
Bands parameter 65
baseline filters 102
bashell folder 17
bashell utility 17
batch processes 11, 17, 91
big data 2
bins 67, 68
Bins parameter 67
BIRT Analytics v, 3
BIRT Analytics Administration 3, 13, 88
BIRT Analytics Administration login page 13
BIRT Analytics Administration tool 102
BIRT Analytics application 102

- BIRT Analytics Loader 2, 3, 4, 102
- BIRT Analytics Loader command-line tool 17
- BIRT Analytics Loader configurations 94
- BIRT Analytics Loader engine 92
- BIRT Analytics Loader features 9, 10
- BIRT Analytics Loader home page 4
- BIRT Analytics Loader login page 4
- BIRT Analytics Loader port 8
- BIRT Analytics Loader user accounts 3
- BIRT Analytics Loader user interface 9, 10, 89
- Boolean values 61
- bubble analysis (defined) 102

C

- calculated columns. *See* calculated fields
- calculated field type 104
- calculated fields 53, 60, 64, 102, 111
- calculated values 111
- Canvas tool 103
- CEIL function 61
- changing
 - column definitions 36
 - data repository 13
 - data source definitions 28
 - data sources 21
 - database names 77
 - database records 13
 - FastDB database 99
 - load processes 28
 - load transactions 30
 - passwords 10, 18
 - project names 9
 - projects 9
 - remote data providers 25
 - scheduled tasks 91
 - transformation scripts 50
 - user functionalities 16
 - user information 89
 - user names 4
- character sets 31
- CHR function 61
- closing
 - BIRT Analytics Administration 16
 - BIRT Analytics Loader sessions 4
 - database tables 37
 - projects 9
- cluster analysis 103
- Clustering tool 103
- CODE function 61
- Code position parameter 55
- column bindings. *See* links
- Column declaration dialog box 33
- column definitions 33, 36, 52
- Column name parameter 72, 73, 76
- column names 55, 76, 77
- column offsets 32
- Column to use parameter 72
- column transformation instructions 53
- column-oriented DBMS 103
- columns
 - See also* calculated fields; fields
 - adding to target tables 33
 - aggregating values for 53
 - changing properties for 36
 - changing values in 53
 - creating calculated 60, 64
 - creating dynamic 45, 53
 - creating parametric 67
 - decoding 55
 - defined 103
 - displaying 43, 52
 - excluding from load processes 34
 - exporting 78
 - extracting from source files 33, 37
 - grouping values for 65, 69
 - linking to 10, 75
 - making permanent 45, 76
 - ranking record subsets and 71
 - removing from tables 34, 73
 - renaming 55, 77
 - sorting values in 58, 72
 - validating 33, 76
 - viewing properties for 43, 44, 52
- command-line options (qLauncher) 92
- command-line tools (BIRT Analytics) 17
- comma-separated values files 24, 34
- comments (project log files) 84
- comments (transformation scripts) 52
- computed columns. *See* calculated fields
- computed fields. *See* calculated fields
- concatenation 60
- concentration (values) 108
- conditional expressions 61
- confidence 103
- configuration parameters 9
 - See also* parameters
- configurations 20, 94
- connection information 20, 21
- connection properties 21, 22, 23
- connections
 - accessing data and 3, 20, 24
 - creating projects and 8
 - running native drivers and 22
 - running non-native drivers and 23
 - setting up 9
 - validating 35
- consequent 103
- constants 61
- control menus 9
- Convert tool 103
- copying
 - FastDB database 98
 - load projects 95
 - repository databases 99
- COUNT function 61
- Count function 54

- count operations 104
- Create new project dialog 7
- creating
 - aggregate columns 53, 54
 - authentication files 17
 - calculated columns 60
 - calculated fields 111
 - data fields 110
 - data sources 21
 - database tables 31
 - decoding columns 55
 - domains 57, 58
 - dynamic columns 45, 53
 - expressions 64
 - FastDB database 97
 - links 47, 75
 - load processes 28–29
 - load transactions 29
 - mapping specifications 29, 30, 37
 - numeric ranges 65
 - parametric columns 67
 - projects 7–8
 - quantile range columns 69, 70
 - queries 37, 38
 - ranking columns 71, 72
 - remote data providers 25
 - reports 3
 - scheduled tasks 90
 - target data tables 33
 - transformation script files 50
 - transformation scripts 51
 - transformation transactions 50
 - user accounts 18
 - users 10, 88
- credential files 91, 92
- credential tokens 92
- credentials 4, 18, 91
- cron 87
- Crosstab analysis (defined) 104
- CSV files 24, 34
- custom data sources. *See* ODBC data sources
- customizing Analytics Loader tabs 10
- cylinder gadgets 104

D

data

- See also* values
- accessing 20
- analyzing 2, 10
- controlling access to 10, 18
- grouping 53, 57, 65, 69
- limiting access to 112
- manipulating 50
- protecting sensitive 3, 10, 91
- retrieving 37, 39
- synchronizing 3, 10, 13, 17
- updating 6
- uploading 25, 27

- viewing 10
- data analysis 104
- data analysis gadgets. *See* gadgets
- Data Explorer 104
- data export operations 78, 79, 80
- data extraction 2, 6, 20, 33
- data field names 55, 77
- data field types 104
- data fields
 - See also* columns
- data fields. *See* fields
- data filters. *See* filters
- data load operations. *See* load operations
- data load processes. *See* load processes
- Data Management module 42
- Data Management tab 10
- data mapping models 29, 30
- data mining 2
- data modeler 6
- data object transformation instructions 73
- data objects 73
- data providers 25, 26
- data repository v, 7, 106
 - See also* FastDB database
- data rows 43, 44, 111
 - See also* records
- data set filters. *See* filters
- data sets 37, 39, 67
- data source connections. *See* connections
- data source definitions 28
- data source mapping specifications 9, 20, 29, 30, 37
- data sources 3, 6, 21
- Data tab (Loader) 9, 20
- data transformation 2, 6, 50
- data transformation rules 9
- Data Tree 105
- data types 34, 44, 105
- database connection information 20, 21
- database connections 20, 22, 23
- database drivers 20, 21
- database loading processes. *See* load processes
- database mapping specifications 37, 38
- database model 105
- Database name parameter 74
- database names 77
- database objects 45, 81
- database repository. *See* FastDB database
- database structures 42
- database tables. *See* tables
- database types 20
- databases 3, 7, 20, 37, 99
 - defined 105
 - See also* data sources
- Databases panel (Data) 20
- Datasource dialog box 22, 23
- date and time field type 104
- date data type 105
- date field type 104

- DATE function 61
- datetime data type 105
- DATETIME function 61
- DAY function 61
- DAYADD function 61
- DAYSTO function 62
- decimal values 34, 60
- Decision Tree 106
- decision tree 105
- Decode dialog box 56
- DECODE instruction 55
- Decode position parameter 55
- Decodes tool 106
- decoding columns 55
- decoding files 55
- decryption 92
- default data repository 7
- deleting
 - columns 34, 73
 - data sources 21, 25
 - links 74
 - load transactions 30
 - scheduled tasks 91
 - script files 51
 - tables 74
 - transformation instructions 52
 - transient links 46, 47
 - user accounts 18
 - users 89
- delimited text files 32
- delimiter characters 32, 55
- Delimiter parameter 55
- Demo database 97
- derived column definitions 52
- derived columns 53
 - See also* calculated fields
- Description parameter (aggregation) 54
- Description property (scheduled tasks) 91
- developers v, 6
- dial gadgets 106
- directory paths 96
- discrete values 44, 52
- Discrete Values Viewer 106
- Discretes tab (Explorer) 44
- displaying
 - columns 43, 52
 - data 10
 - database tables 43
 - databases 42
 - discrete values 44, 52
 - load process log files 9, 13, 91
 - project log files 12
 - project names 8, 9
 - script files 84
 - transformation instructions 51
 - user functionalities 16
- DISTINCT function 62
- documentation v

- Domain definition dialog box 58
- DOMAIN instruction 57
- domain names 58
- Domain parameter 58
- domains 57, 78
- Downloads tool 106
- driver names 23
- drivers 3, 20, 21, 23
- DROPCOLUMN instruction 73
- DROPDATABASE instruction 73
- DROPLINK instruction 74
- DROPTABLE instruction 74
- Dubnium database 4, 94
- Dubnium servers 95, 97
- Dubnium service 3
- Dubnium service security accounts 3, 18
- dubnium.exe 106
- dynamic columns 45, 53, 76

E

- edit panel (table declarations) 37
- editable columns 45
- editor (Transform) 53
- Encoding parameter 55
- encoding types 31
- encrypted authentication files 17
- encryption 92
- End line parameter 55
- end-of-line characters 32, 55
- EQ operator 61
- estimation method 107
- ETL (defined) 2
- ETL developers v, 6
- ETL processes 2, 6
 - See also* specific process
- events 101
- evolution analysis (defined) 106
- EXP function 62
- Explorer 42–45
- Export Analytic DB tool 106
- Export dialog box 78, 79, 80
- Export File tool 106
- EXPORT instruction 78
- export operations 78, 79, 80
- Export target and name parameter 78
- export tools 106
- Expression dialog box 64
- EXPRESSION instruction 60
- Expression parameter 60
- expressions 60, 64
 - defined 106
- External Call page (Administration) 91
- extraction processes 2, 6, 20, 33

F

- FastDB database
 - accessing files for 95
 - adding databases to 99

- backing up 96
- changing 13, 99
- copying 98
- creating 97
- installing 96
- loading data into 7
- removing databases from 73
- removing objects from 45
- renaming data fields in 55, 77
- renaming databases in 77
- running 3
- setting up security accounts for 18
- testing for objects in 81
- viewing database structures in 42
- FastDB engine 3
- FastDB engine user accounts 3, 10, 18
- FastDB web service 107
- features 9, 10
- field (defined) 104
- field names 55, 77
- field types 104
- fields
 - adding calculated 111
 - building summary tables and 108
 - creating 110
 - See also* columns
- file connections 24
- file encoding types 31
- file headers 32
- file lists 27
- file names 27
- File parameter 55
- file paths 96
- file systems 25
- file types 31
- File Upload dialog box 25
- files
 - accessing data in 24
 - copying project 95
 - creating authentication 17
 - creating script 50
 - data extraction and multiple 27
 - defining mapping specifications for 30, 34
 - exporting data to 78, 79, 80
 - generating credential 91
 - opening script 51
 - selecting formatting properties for 31
 - transforming data and 50
 - uploading 25, 26, 27, 55
 - viewing log 9, 12, 13, 91
- Files panel (Data) 20, 25
- Filter parameter
 - AGGREGATE instruction 54
 - EXPORT instruction 78
 - INUMERIC instruction 65
 - QUANTILE instruction 70
 - RANKING instruction 72
- filter types 107
 - See also* specific type
- filtering
 - record groups 102
- filters 107
- final links 46
- First function 54
- flat file data sources 25
- flat files 20, 32
 - See also* data sources
- FLOOR function 62
- Force sequence parameter 72
- Forecasting tool 107
- formatting properties (source files) 31
- formula names 45
- formulas 45, 53
 - See also* expressions
- full numeric field type 104
 - See also* numeric fields
- Function parameter 53
- functionalities 16, 107
- functions 53, 61
- funnel gadgets 107

G

- gadget types 107
- gadgets 107
- Gallery 107
- GE operator 61
- geographic maps 109
- grouping
 - records 112
- grouping data 53, 57, 65, 69
- groups (users) 13, 107
- GT operator 61

H

- has seasonality option 107
- help (qLauncher) 92
- Holt-Winters method 107
- home page (Loader) 4
- HOUR function 62

I

- IF function 62
- If statements 61
- IFEXISTS instruction 81
- IFLINKED instruction 82
- IIS services 10, 11
- import operations 108
- importing metadata 36
- inactive transformation instructions 66
- Include dialog box 84
- INCLUDE instruction 83
- INCLUDE statements 81, 83
- Includes panel (Transform) 50
- indexed fields 108
 - See also* summary tables

- indexes 33, 44
- information. *See* data
- INSTR function 62
- INT function 62
- integer data type 105
- Internet Information Services. *See* IIS services
- INUMERIC instruction 65
- inversion 58
- IP addresses 3, 8
- IP parameter 8
- ISNOTNULL function 62
- ISNULL function 62

K

- key pairs (encryption) 92
- Key to rank parameter 71
- KURT function 62
- Kurtosis 108
- Kurtosis coefficient 108

L

- label gadgets 108
- Language property 89
- Last function 54
- LCASE function 62
- LE operator 61
- LEFT function 62
- LEN function 62
- leverage 108
- lift 108
- Link dialog box 75
- LINK instruction 75
- links 74, 75, 82
- Links tool 42, 46–47, 108
- load operation types 31
- load processes
 - automating 92
 - building mapping models for 29, 30
 - building projects for 3, 6, 7
 - configuring 20
 - creating 28–29
 - defining connections for 20, 21, 24
 - excluding transactions from 31
 - running 2, 9
 - saving changes to 28
 - scheduling 4, 89
 - setting options for 29
 - starting 10
 - viewing log files for 9, 91
- load projects. *See* projects
- load transactions
 - creating 29
 - defining source file formats for 31, 34
 - defining table columns for 33
 - defining target objects for 30, 37, 38
 - excluding from load processes 31
 - loading data and 20, 28, 39
 - running 29

- Loader module 102
- Loader module. *See* BIRT Analytics Loader
- loading processes. *See* load processes
- local file systems 25
- log files 9, 12, 13, 84, 91
- LOG function 62
- LOG10 function 62
- logging load operations 29
- login credentials 4
- login names 89
- login page (Analytics Administration) 13
- login page (Analytics Loader) 4
- Login property 89
- Logout icon (Loader) 4, 9
- Longint data type 105
- LONGINT function 62
- LT operator 61
- LTRIM function 62

M

- Main file link 50, 51
- make permanent operations 109
- MAKEPERMANENT instruction 76
- map analysis (defined) 109
- mapping 29
- mapping specifications 9, 20, 29, 30, 37
- mathematical operators 60
- MAX function 62
- Maximum value function 54
- maximum values 109
- MEAN function 62
- Mean function 54
- mean values 109
- MEDIAN function 62
- median values 109
- menus 9
- metadata 36
- meter gadgets 109
- Microsoft Internet Information Services. *See* IIS services
- Microsoft SQL Server databases 20
- MID function 63
- MIN function 63
- Minimum value function 54
- minimum values 109
- MINUTE function 63
- MODE function 63
- mode values 109
- MONTH function 63
- My Data Viewer 109
- My Folders Viewer 109
- MySQL databases 20, 21

N

- Name parameter 8
- naming
 - database connections 21, 22
 - decoding columns 56

- domains 58
- FastDB data fields 55, 77
- file data sources 27
- ODBC database connections 23
- parametric columns 68
- projects 8
- quantile range columns 70
- ranking columns 72
- variables 85
- native drivers 20, 21
- NE operator 61
- negative values 112
- NetScaler Web Logging queries 110
- New file list dialog box 27
- non-native drivers 21
- normal distribution 110
- NOT operator 61
- NOW function 63
- NSWL queries 110
- null values 44
- Number of parts parameter 70
- numeric field types 104
- numeric ranges 65, 66
- Numeric ranges dialog box 66
- Numeric Ranges tool 110
- numeric values 65, 69
- numerical estimation method. *See* Holt-Winters method

O

- ODBC connectors 3
- ODBC data sources 23
- ODBC driver names 23
- ODBC drivers 21, 23
- online documentation v
- opening
 - BIRT Analytics Admin application 13, 88
 - BIRT Analytics Loader 3
 - BIRT Analytics Loader projects 8
 - Dubnium database 94
 - script files 51
 - Transform text editor 53
- operators 60, 90
- Options panel (Data) 20, 29
- OR operator 61
- Oracle databases 20, 38
- output files (qLauncher) 94

P

- parameters
 - adding decoding columns and 55
 - adding parametric columns and 67
 - adding quantile range columns and 69
 - adding ranking columns and 71
 - aggregating data values and 53
 - changing project configurations and 9
 - creating domains and 57
 - creating expressions and 60

- creating projects and 8
- defined 110
- defining numeric ranges and 65
- defining variables and 85
- dropping database objects and 73, 74
- dropping links and 74
- entering values for 111
- exporting domains and 78
- linking tables and 75
- opening Loader module and 3
- running bashell utility and 17
- running INCLUDE statements and 82
- running qLauncher utility and 92
- testing for links and 82
- uploading remote files and 26
- parametric columns 67
- Parametric dialog box 67, 68
- parametric filters 110
- PARAMETRIC instruction 67
- Parametric tool 110
- Pareto analysis (defined) 110
- Pareto principle 110
- Password parameter 8
- Password property 89
- passwords 4, 10, 18, 89
- paths 96
- periodic trends 112
- permanent columns 45, 76
- permissions. *See* privileges; functionalities
- Port parameter 8
- positive values 112
- PostgreSQL databases 20
- POWER function 63
- predictive analytics 110
- private encryption keys 92
- privileges 10, 13, 97, 101, 107
- processes. *See* ETL processes
- profile (users) 111
- profile analysis (defined) 110
- project development page 9
- project files 95
- project folders 95
- project log files 12, 84
- project names 8, 9
- Project property 90
- projects
 - building ETL processes and 3, 6
 - creating 7
 - limiting log entries for 12
 - managing 9
 - naming 8
 - running 12, 92, 93
 - saving 9, 37
 - storing 95
- Projects management page 8, 88
- prompted filter 111
- PROPER function 63
- PROPERSENTENCE function 63

- protecting data 3, 10, 91
- public encryption keys 92
- public-key encryption 92

Q

- qLauncher utility 92, 93
- qualifier characters 33, 55
- Qualifier parameter 55
- Quantile dialog box 70
- QUANTILE instruction 69
- quantile range columns 69
- quantile range filters 70
- Quantile tool 111
- QUARTER function 63
- queries
 - creating 37, 38
 - defined 111
 - defining bins and 68
 - defining domains and 58
 - limiting data access and 112
 - retrieving data and 37, 67
 - retrieving data with 111
 - tracking HTTP data traffic and 110
- Query columns parameter 78
- query language 112
- Query resolution table parameter 78

R

- RAND function 63
- range of values 65, 66
- ranking 71
- ranking columns 71
- Ranking dialog box 72
- RANKING instruction 71
- ranking keys 71
- ranking sequence 72
- Ranking tool 111
- real data type 105
- REAL function 63
- real numeric field type 104
 - See also* numeric fields
- records
 - adding to tables 31
 - changing 13
 - creating sample of 58
 - defined 111
 - grouping 57, 112
 - linking tables and 46
 - ranking subsets of 71
 - setting number of 29
 - updating 13
 - validating content 33
- relational databases 20
- remote data providers 25, 26
- removing
 - columns 34, 73
 - data sources 21, 25
 - links 74

- load transactions 30
- scheduled tasks 91
- script files 51
- tables 74
- transformation instructions 52
- transient links 46, 47
- user accounts 18
- users 89
- RENAME instruction 77
- renaming
 - databases 77
 - FastDB database fields 55, 77
 - projects 9
 - SYSTEM user 4
- REPLACE function 63
- REPORT instruction 84
- reports 3, 13
- repository v, 7, 106
 - See also* FastDB database
- Repository list (Transform) 52
- Repository tree 42, 43
- resolution 58, 74
- restarting IIS services 11
- RIGHT function 63
- roles 112, 114
- ROW function 63
- rows 43, 44, 111
 - See also* records
- RTRIM function 63
- rules (data transformation) 9
- running
 - Analytics Loader projects 12, 91, 92, 93
 - bashell utility 17
 - load processes 2, 9
 - load transactions 29
 - transformation instructions 52, 66
 - transformation scripts 50, 82, 85
- Running project log 12

S

- saving
 - authentication information 17
 - load processes 28
 - projects 9, 37
- Scheduled task dialog 90
- scheduled tasks 111
- Scheduled tasks page (Administration) 89
- scheduling load processes 4, 89
- Scratchpad 112
- script files 50, 51, 83
- scripts. *See* transformation scripts
- seasonal patterns 112
- seasonal periodicity 112
- seasonal trends 107
- seasonality 112
- SECOND function 63
- SECSTO function 63
- secure connections 8

- security 3, 10, 91
- security accounts (Dubnium service) 3, 18
- security filters 112
- security groups 13
- security keys 92
- Security module (Analytics Admin) 14
- Security page (Analytics Admin) 14
- security roles 112
- segment blocks 58
- segments 57
 - defined 112
- selections 112
- server names 3
- servers 2, 10
- Services utility 11
- Settings property 90
- Settings tab (Loader) 9
- SGN function 63
- sharing connections 20
- SKEW function 63
- skewness operations 112
- skewness values 112
- Sorting direction parameter 72
- Source column parameter 53, 55, 65, 69
- source files 31, 33
- sphere gadgets 112
- spreadsheets 24
- SQL (defined) 112
- SQL queries. *See* queries
- SQL Server databases 20
- sqlite database 4, 94
- SQLite Manager 94
- SQRT function 64
- Square sum function 54
- squared values 113
- SSL parameter 8
- stage 112
- standard deviation 113
- Standard deviation function 54
- standard normal distribution 113
- Standardize column tool 113
- starting
 - IIS services 11
 - load processes 10
 - synchronization processes 13, 17
- STDEV function 64
- stopping
 - Dubnium servers 95
 - IIS services 11
- string data type 105
- STRING function 64
- STRIP function 64
- sum (defined) 113
- SUM function 64
- Sum function 54
- summary values 113
- sum-of-squares (defined) 113
- SUMSQ function 64

- support 113
- SWITCH statements 61
- symmetric distribution 112
- Synchronize option 14
- Synchronize page (Analytics Admin) 14
- synchronizing data 3, 10, 13, 17
- system administration tool. *See* BIRT Analytics Administration
- system administrators. *See* administrators
- system privileges 107
- SYSTEM user 4

T

- Table declaration dialog box 30, 33, 37
- table definitions 6, 9, 28, 30
- Table name parameter 74
- tables
 - adding calculated columns to 53
 - adding dynamic columns to 53
 - adding fields to 110
 - binding columns to. *See* links
 - closing 37
 - creating 31
 - defined 113
 - deleting 74
 - exporting 78
 - grouping data from multiple 53
 - grouping records in. *See* segments
 - linking to 42, 46–47, 74, 75
 - mapping source file data and 30
 - ranking 71
 - removing columns in 34, 73
 - renaming 77
 - selecting 37
 - setting sorting direction for 72
 - testing for links in 82, 83
 - updating 6, 13
 - viewing columns in 43, 52
 - viewing list of 43
 - viewing properties for 43
- tabs 9, 10
- Target column parameter
 - AGGREGATE instruction 53
 - DECODE instruction 55
 - DOMAIN instruction 57
 - EXPRESSION instruction 60
 - INUMERIC instruction 65
 - PARAMETRIC instruction 67
 - QUANTILE instruction 70
- target filters 113
- target objects 29, 30, 37
- Target table parameter
 - AGGREGATE instruction 53
 - DOMAIN instruction 57
 - EXPRESSION instruction 60
 - PARAMETRIC instruction 67
- temporal files 113
- temporary files 113

- text editor (Transform) 53
- text field type 104
- text file data sources 24
- text files 32, 79
 - See also* data sources
- Text parameter 84
- time data type 105
- time field type 104
- TIME function 64
- time settings (scheduled tasks) 90
- TODAY function 64
- tokens (credentials) 92
- tools menu 10
- transactions 50
 - See also* load transactions
- Transform tab (Loader) 9, 50
- transformation instructions 51, 66, 77
 - See also* transformation scripts
- transformation processes 2, 6, 50
- transformation rules 9
- transformation script files 50, 51, 83
- transformation scripts 29, 50, 51, 82, 85
- transformation transactions 50
- transient files 113
- transient links 46
- triggers 113
- TRIM function 64

U

- UCASE function 64
- Unicode data type 105
- universal filters 114
- updating
 - database table definitions 28
 - database tables 6, 13
 - Includes file list 51
- uploading
 - data 25, 27
 - decoding files 55
 - local files 25
 - multiple files 27
 - remote files 26
- URI dialog box 26
- URI property 26
- URIs 26
- URLs
 - BIRT Analytics Administration 13
 - BIRT Analytics Loader 3
- user accounts 4, 10, 18, 89
- user credentials 4, 18, 91
- user groups 13, 107
- user interface 9, 10, 89
- user names 4
- User parameter 8
- user privileges 10, 13, 97
- users 3, 10, 88
- user-specified requests. *See* selections

V

- VAL function 64
- values
 - aggregating 53
 - defining range of 65, 66
 - getting from multiple tables 67
 - grouping 65, 69
 - inverting 58
 - loading 45
 - ranking 111
 - viewing discrete 44, 52
 - viewing number of discrete 44
 - viewing number of null 44
- VAR instruction 85
- Variable dialog box 85
- variables 85
- variance 114
- Venn analysis (defined) 114
- viewers 109
- viewing
 - columns 43, 52
 - data 10
 - database tables 43
 - databases 42
 - discrete values 44, 52
 - load process log files 9, 13, 91
 - project log files 12
 - project names 8, 9
 - script files 84
 - transformation instructions 51
 - user functionalities 16

W

- web addresses 26
- web browsers 3, 10
- web servers 3, 10
- web services v
- web services. *See* BIRT Analytics application; FastDB
 - web service
- WEEKDAY function 64
- workflow 114
- workflow roles 114

Y

- YEAR function 64

Z

- zero values 112
- Z-score operations 114