



Data Access Guide

Copyright

© 2008 Business Objects. All rights reserved. Business Objects owns the following U.S. patents, which may cover products that are offered and licensed by Business Objects: 5,555,403; 5,857,205; 6,289,352; 6,247,008; 6,490,593; 6,578,027; 6,831,668; 6,768,986; 6,772,409; 6,882,998; 7,139,766; 7,299,419; 7,194,465; 7,222,130; 7,181,440 and 7,181,435. Business Objects and the Business Objects logo, BusinessObjects, Business Objects Crystal Vision, Business Process On Demand, BusinessQuery, Crystal Analysis, Crystal Applications, Crystal Decisions, Crystal Enterprise, Crystal Insider, Crystal Reports, Desktop Intelligence, Inxight, the Inxight Logo, LinguistX, Star Tree, Table Lens, ThingFinder, Timewall, Let there be light, Metify, NSite, Rapid Marts, RapidMarts, the Spectrum Design, Web Intelligence, Workmail and Xcelsius are trademarks or registered trademarks in the United States and/or other countries of Business Objects and/or affiliated companies. All other names mentioned herein may be trademarks of their respective owners.

Third-party Contributors

Business Objects products in this release may contain redistributions of software licensed from third-party contributors. Some of these individual components may also be available under alternative licenses. A partial listing of third-party contributors that have requested or permitted acknowledgments, as well as required notices, can be found at: <http://www.businessobjects.com/thirdparty>

Contents

Chapter 1	Data access basics	9
	About Connection Server.....	10
	Components of a connection.....	10
	System architecture.....	11
	About data access drivers.....	11
	About data access configuration files.....	12
	Global configuration files.....	13
	Driver configuration files.....	13
Chapter 2	Creating a connection	15
	Before you create a connection.....	16
	Checking connection configuration: cscheck.....	16
	Displaying help on the cscheck tool.....	17
	Running the cscheck tool.....	18
	Check tool—function overview.....	18
	Check tool—list.....	19
	Check tool—drivers search.....	20
	Check tool—find.....	21
	Check tool—middleware.....	22
	Check tool—accessdriver.....	23
	Check tool—connectivity.....	24
	Check tool—ping	25
	Accessing the New Connection wizard from the Designer Connections list.....	27
	Accessing the New Connection wizard from the Parameters dialog box...	28
	Using the New Connection Wizard.....	28

Contents

Creating a new connection	29
Database Middleware Selection dialog box	30
Login Parameters dialog box.....	32
Configuration parameters dialog box	34
Custom Parameters dialog box.....	35
About JDBC connections	36
Creating a JDBC connection	36
JDBC SBO example file structure.....	37
About JavaBean connections.....	38
Creating a JavaBean connection.....	38
JavaBean SBO example file structure	39
Chapter 3 Configuring data access global parameters	41
About global parameters.....	42
About the cs.cfg global configuration file.....	42
Viewing and editing cs.cfg.....	43
Configuring the <driver defaults> parameters.....	43
Configuring <Settings> parameters.....	44
Charset List Extension.....	44
Config File Extension.....	45
Description Extension.....	45
Enable Failed Load.....	45
Load Drivers On Startup.....	46
Max Pool Time.....	47
SQL External Extension.....	48
SQL Parameter Extension.....	48
Strategies Extension.....	48
Configuring CORBA access.....	49
Configuring the Distribution section for CORBA access.....	50

Chapter 4	Configuring data access driver parameters	51
	Configuring driver parameters.....	52
	Data access configuration files.....	52
	Installed SBO files.....	53
	Editing an SBO file.....	54
Chapter 5	SBO file parameter reference	55
	SBO parameter categories.....	56
	SBO file structure.....	56
	Default SBO parameters.....	57
	Informix SBO parameters.....	75
	JavaBean SBO parameters.....	77
	JDBC SBO parameters.....	78
	ODBC SBO parameters.....	79
	ODBC3 SBO parameters.....	81
	OLE DB SBO parameters.....	82
	Sybase ASE/CTLIB.....	83
Chapter 6	Configuring SQL generation parameters for a universe	85
	About SQL generation parameters for a universe.....	86
	Editing SQL generation parameters in a universe.....	86
	Universe SQL parameters reference.....	87
Chapter 7	Configuring SQL generation parameters for a database	105
	About SQL generation parameters.....	106
	About PRM files.....	106
	Parameter file structure.....	108
	Viewing and editing PRM file parameters.....	109
	Viewing and editing function help text.....	110

Contents

PRM file Configuration reference.....	111
BACK_QUOTE_SUPPORTED.....	112
CASE_SENSITIVE.....	112
CHECK_OWNER_STATE.....	112
CHECK_QUALIFIER_STATE.....	113
COMMA.....	113
CONCAT.....	114
CONSTANT_SAMPLING_SUPPORTED.....	114
DATABASE_DATE_FORMAT.....	115
DATATYPE_BLOB.....	115
DATATYPE_DOUBLE.....	115
DATATYPE_DTM.....	116
DATATYPE_INT.....	116
DATATYPE_NULL.....	116
DATATYPE_STRING.....	116
DATE_WITHOUT_QUOTE.....	117
EXT_JOIN.....	117
EXT_JOIN_INVERT.....	118
EXTERN_SORT_EXCLUDE_DISTINCT.....	118
GROUPBY_EXCLUDE_COMPLEX.....	119
GROUPBY_WITH_ALIAS.....	119
GROUPBY_WITHOUT_CONSTANT.....	120
GROUPBYCOL.....	120
IDENTIFIER_DELIMITER.....	120
IF_NULL.....	121
INTERSECT.....	121
KEY_INFO_SUPPORTED.....	122
LEFT_OUTER.....	122
LENMAXFORCOLUMNNAME.....	123
LENMAXFORTABLENAME.....	123
LENMAXFORVARCHAR.....	123

Contents

MINUS.....	124
NO_DISTINCT.....	124
NULL_IN_SELECT_SUPPORTED.....	125
OLAP_CLAUSE.....	125
OUTERJOINS_GENERATION.....	126
OVER_CLAUSE.....	129
OWNER.....	129
PERCENT_RANK_SUPPORTED.....	130
PREFIX_SYS_TABLE.....	130
QUALIFIER.....	131
QUOTE_OWNER.....	131
RANK_SUPPORTED.....	132
REFRESH_COLUMNS_TYPE.....	132
REVERSE_TABLE_WEIGHT.....	132
RIGHT_OUTER.....	133
RISQL_FUNCTIONS.....	133
SEED_SAMPLING_SUPPORTED.....	134
SORT_BY_NO.....	135
UNICODE_PATTERN.....	135
UNION.....	135
USER_INPUT_DATE_FORMAT.....	136
USER_INPUT_NUMERIC_SEPARATOR.....	137

Chapter 8 Data type conversion reference 139

Data type conversion	140
IBM DB2.....	140
Informix.....	141
Microsoft SQL Server.....	142
Oracle.....	143
Red Brick.....	144
TIME and TIMESTAMP support in a WHERE clause.....	145

Contents

	Sybase.....	145
	Teradata.....	146
Chapter 9	Business Objects information resources	149
	Documentation and support.....	150
	Useful addresses at a glance.....	150
	Documentation.....	151
	What's in the documentation set?.....	151
	Where is the documentation?.....	151
	Send us your feedback.....	152
	Customer support, consulting and training.....	152
	How can we support you?.....	152
	Looking for the best deployment solution for your company?.....	153
	Looking for training options?.....	153
Appendix A	Get More Help	155
Index		159



Data access basics



1

chapter



About Connection Server

Connection Server is the Business Objects data access software that manages the connection between the application and the data source.

Connection Server allows Business Objects applications such as Designer and Web Intelligence users to connect to and run queries against a data source.

Connection Server does not have a user interface. You create and administer connections from the user interface of Business Objects applications such as Designer, or by editing the configuration files.

- **Creating connections:** You create connections using a connection wizard available from Business Objects applications such as Designer. You can modify some connection parameters from the application.
- **Optimizing data access:** You can optimize the way that data is passed through Connection Server by modifying data access parameter files and SQL generation files. These files are in XML format, and are installed with Connection Server. You can set parameter values for either a specific driver, or for all installed data access drivers.

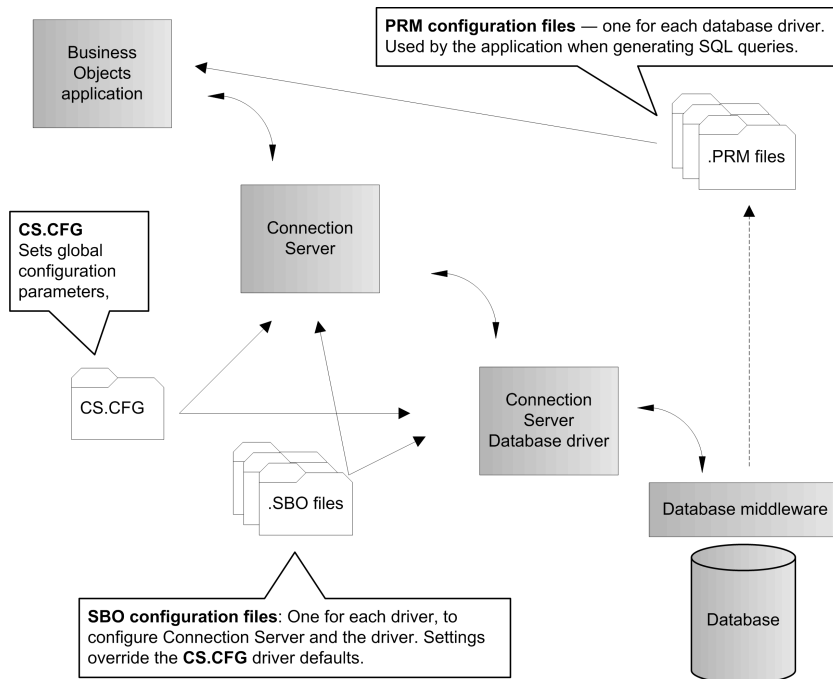
Components of a connection

Connection Server manages the data access connection between a Business Objects application and a target database. A Business Objects data access connection consists of the following components:

- Connection Server manages the connection between the Business Objects application and the data source. For example, it handles requests for data from the application.
- A data access driver is database-specific software that manages the connection between Connection Server and the database middleware.
- Configuration files define parameters to configure the connection between:
 - the Business Objects application and Connection Server
 - the Business Objects application and the data access driver
 - Connection Server and the data access driver

System architecture

The diagram below details where Connection Server and data access drivers fit into a Business Objects configuration.



About data access drivers

Data access drivers provide the connection between Connection Server and a data source. A database requires a data access driver in order that it can be accessed by a Business Objects application.

Business Objects applications include data access drivers that you can use to configure connections to your databases. The data access drivers that are included can depend on your licence. You can also develop your own data access drivers for use with non-standard databases.

To configure a database connection for which you do not have a driver, you need to obtain the required drivers before you can create the connection. The following options are available for obtaining a driver:

- Contact your Business Objects representative to determine if there is a driver available, and if you are licenced to use it.
- Use the Driver Development Kit (DDK) to develop a driver to use. Contact your Business Objects representative for details.

When you create a new connection, you select the appropriate data access driver for the target datasource. For example, if you access an Oracle 10g database, you must install the appropriate middleware (Oracle 10g Client), then the Business Objects Oracle data access driver.

For an up-to-date list of supported data access drivers, check the Business Objects Support Site at www.support.businessobjects.com, or contact your BusinessObjects representative.

About data access configuration files

Data access configuration files are installed when you install Business Objects products. The configuration files can be divided into two levels:

- Global level: These configuration files apply to all connections.
- Driver level: These configuration files apply to specific drivers.

Of the files described in this section:

- Some of the files contain parameters that you can modify to optimize or customize data access.
- Other files listed must not be modified. They are described purely to explain their function.

In addition to the configuration files that control a connection, each data access driver has a `.prm` configuration file. These files are used by Business Objects applications such as Designer. They control the way in which the Universe generates SQL. This functionality allows you to configure Universe-specific SQL generation properties.

Note: `connectionserver-install-dir` variable in this guide

In this guide, the variable `connectionserver-install-dir` is the install root path for the data access files used by Business Objects applications.

Under Windows, the default *connectionserver-install-dir* = :C:\Program Files\Business Objects\BusinessObjects Enterprise 12.0\win32_x86\dataAccess

Global configuration files

The *cs.cfg* global configuration file files used by all data access drivers is installed in the following location:

- *connectionserver-install-dir*\connectionServer

The *cs.cfg* file contains parameters that apply to all installed data access drivers.

Related Topics

- [About global parameters](#) on page 42

Driver configuration files

The configuration files used by data access drivers are installed in the following path:

- On a Windows system:

connectionserver-install-dir \connectionServer*RDBMS*

- On a Unix system:

connectionserver-install-dir /*RDBMS*/connectionServer


where *RDBMS* is the name of the database technology that uses the configuration file.

The files listed below have parameters that apply to installed data access drivers.

Driver specific file	Can be edited?	Description	Example
<driver> .sbo	Yes	There is a .sbo file for each supported network protocol, or database middle-ware used to connect to a database. Defines the specific connectivity configuration for each database.	oracle.sbo
<driver> .prm	Yes	Defines parameters that affect the way that a Business Objects application generates SQL.	oracle.prm
<driver><language> .cod	No	Stores information related to connection definitions. Defines the fields that appear when you create a new connection. Note: Do not modify these files.	oracleen.cod

Related Topics

- [SBO parameter categories](#) on page 56
- [PRM file Configuration reference](#) on page 111
- [About SQL generation parameters](#) on page 106



Creating a connection



2

chapter



Before you create a connection

This section lists the things you need to do before you can create a connection.

- Ensure that your platform conforms to the platforms supported for Business Objects connections.
- Ensure that the database middleware is installed correctly, and that you can access your database through either your computer or a server.

You can use the `cscheck` tool to check your configuration. See the `cscheck` tool documentation for details.

- Ensure that you have all the information necessary to access your database, for example the database login name and password.
- Install the Business Objects product that you will use, including the appropriate data access driver.
- Check that all of the Business Objects services have started successfully.
- Refer to the Readme notice that comes with your Business Objects product to check for any configuration changes that your environment and software might require.

Related Topics

- [Checking connection configuration: `cscheck`](#) on page 16

Checking connection configuration: `cscheck`

The Connection Server software includes a command line utility that you can use to check your datasource connection infrastructure. You can use the `cscheck` tool to check your client middleware and the installed Business Objects data access drivers at any time.

Note: The results of all checks apply to your local machine, from which you run the tool.

The `cscheck` tool is installed in the following directory: `connectionserver-install-dir/connectionServer/tools`

You run the `cscheck` tool from a command console (DOS or shell). The output is displayed on the screen. You can specify that the output is generated as XML format, or you can suppress output, to use the tool in a script.

The `cscheck` tool can perform the following functions on your local machine:

- Return details of all the connectivities, that is network layers and databases, that the installation can support.
- Return details of the data access drivers installed on your local machine.
- Return details of the connectivities installed on your local machine.
- Check for a valid middleware installation for a supplied network layer and database client.
- Check for a valid data access driver installation for a supplied network layer and database client.
- Check if a connection can be established to a given database.

Related Topics

- [Check tool—function overview](#) on page 18

Displaying help on the `cscheck` tool

The `cscheck` tool provides functionality to:

- Display general help on the `cscheck` utility.
- Display help on each available `cscheck` function.

The help can be displayed in any language selected when you installed your Business Objects applications.

To display general help on `cscheck`, use the following syntax:

Figure 2-1: Command help syntax

`cscheck --help|h --language|l { language }`

To display help on a function, use the following syntax, where **functionName** is the name of the function for which you want help, and **language** is the language in which to display the help:

Figure 2-2: Function help syntax

`cscheck --help|h { functionName } --language|l { language }`

Example:

To display help in English on the `cscheck` tool, use the following command:

```
cscheck --help
```

To display help in French on the `connectivity` function, use the following command:

```
cscheck --language fr --help connectivity
```

Running the `cscheck` tool

You can run the `cscheck` tool at any time after you have installed your Business Objects application software.

1. Open a command console.
2. Change directory to the path where the tool is installed.
3. Enter `cscheck` with the correct parameters to find the information that you want.
4. Review the returned information.

Check tool — function overview

From a command console, you use the `cscheck` command with the appropriate function and its arguments to return the results that you want.

`cscheck` commands have the following structure. Some of the parameters are optional.

Figure 2-3: `cscheck` syntax

`cscheck --language|l { output language }--xml|x--mute|m function namefunction options`

The first part of the command controls the output format:

- `<output language>` or `l` followed by the language specified in ISO-639 standard. This is optional. The default language is English.
- `--xml` or `x` specifies that the output is in XML format. This is optional. The default output is text displayed on the screen.

- `--mute` specifies that the output is not generated. You would use this switch if you were using the tool in a script that checked the returned status. This is optional. The default is that output is generated.

The remaining part of the command consists of the function and its option arguments.

<function name> can take the following values. Each function has a short version that you can use in place of the full function name:

- `list`, or `lt`
- `driverssearch`, or `ds`
- `find`, or `fd`
- `middleware`, or `mw`
- `accessdriver`, or `ad`
- `connectivity`, or `ct`
- `ping`, or `pg`

Related Topics

- [Check tool—accessdriver](#) on page 23
- [Check tool—connectivity](#) on page 24
- [Check tool—drivers search](#) on page 20
- [Check tool—find](#) on page 21
- [Check tool—list](#) on page 19
- [Check tool—middleware](#) on page 22
- [Check tool—ping](#) on page 25

Check tool—list

This function returns a list of the supported network layers and database engines. For example you could use it to determine the correct values to use with other check tool functions.

Note: This function returns the full list of supported data access drivers and middleware, including those that are not necessarily installed on your machine.

Figure 2-4: list syntax

cscheck |list||lt|

Example:

The following command lists all network layers and database engines supported by the Business Objects installation on the current machine.

```
cscheck list
```

Related Topics

- [Check tool—function overview](#) on page 18
- [Displaying help on the cscheck tool](#) on page 17

Check tool — drivers search

This function returns a list of the installed data access drivers.

Figure 2-5: driverssearch syntax

cscheck |driverssearch||ds|

Example:

The following command lists all data access drivers installed on the machine.

```
cscheck driverssearch
```

Related Topics

- [Check tool—function overview](#) on page 18
- [Displaying help on the cscheck tool](#) on page 17

Check tool – find

This function lists the available connectivity types, that is middleware and database clients, that are available from the local machine. This includes:

- Connectivity types available on the local machine
- Connectivity types available using the CORBA communication layer
- Connectivity types available using the HTTP communication layer
- Java connectivity types available on the local machine

Figure 2-6: *find* syntax

cscheck |**find**|**fd** |**-m** { *Connection Server access mode* }

Table 2-1: *Function input parameters*

Connection Server access mode (<i>-m</i>)	<p>The mode in which the client application accesses Connection Server:</p> <ul style="list-style-type: none"> • local: lists connectivity types available on the local machine. • corba: lists connectivity types available using CORBA. • http: lists connectivity types available using HTTP. • java: lists Java connectivity types available on the local machine. • extended: lists local, java, and corba connectivity types.
---	--

Example: Finding local connectivities

The following command returns a list of the data access drivers on the local machine that can be loaded by Connection Server.

```
cscheck find -m local
```

Example: Finding CORBA server connectivities

The following command returns a list of the data access drivers available from a CORBA server.

```
cscheck find -m corba
```

Related Topics

- [Check tool—function overview](#) on page 18
- [Displaying help on the *cscheck* tool](#) on page 17

Check tool — middleware

For a supplied network layer and database client, this function checks for a valid installation of the client middleware. To check both the middleware and data access driver for a supplied network layer and database client, you can use the `connectivity` function.

Figure 2-7: middleware syntax

cscheck [middleware][mw]-c { network layer }-d{ database client }

Table 2-2: Function input parameters

network layer (-c)	The network layer that the database middleware uses, as returned by the <code>find</code> function.
database client (-d)	The database to check, as returned by the <code>find</code> function.

Example:

The following command checks for a valid installation of the Oracle Client 9 middleware on the local machine. It creates an XML file of the output:

```
c:\result.xml
```

```
cscheck --xml middleware -c "Oracle Client" -d "Oracle 9" >
c:\result.xml
```

Related Topics

- [Check tool—function overview](#) on page 18
- [Displaying help on the cscheck tool](#) on page 17
- [Check tool—connectivity](#) on page 24
- [Check tool—accessdriver](#) on page 23

Check tool — accessdriver

For a supplied network layer and database client, this function checks for a valid data access driver installation. To check both the middleware and data access driver for a supplied network layer and database client, you can use the `connectivity` function.

Figure 2-8: accessdriver syntax

cscheck [**accessdriver**][**ad**]-**c**{ *network layer* }-**d**{ *database client* }

Table 2-3: Function input parameters

network layer (<i>-c</i>)	The network layer that the database middleware uses, as returned by the <code>find</code> function.
database client (<i>-d</i>)	The database to check, as returned by the <code>find</code> function.

Example:

The following command checks for a valid installation of an Oracle 9 data access driver, and displays the output in French:

```
cscheck -l fr accessdriver -c "Oracle Client" -d "Oracle 9"
```

Related Topics

- [Check tool—function overview](#) on page 18
- [Displaying help on the *cscheck* tool](#) on page 17
- [Check tool—list](#) on page 19

Check tool—connectivity

For the supplied network layer and database client, this function checks that both the installed middleware and the data access driver are valid.

You can check each individually using the `middleware` and the `accessdriver` functions. You can use the `ping` function to check if you can connect to a specific database.

Figure 2-9: connectivity syntax

cscheck |connectivity|ct|-c{ network layer }-d{ database client }

Table 2-4: Function input parameters

network layer (-c)	The network layer that the database middleware uses, as returned by the <code>find</code> function.
database client (-d)	The database to check, as returned by the <code>find</code> function.

Example:

The following command checks the installed Oracle client middleware, and the Oracle 9 data access driver. The command writes the output to a text file: `c:\result.txt` .

```
cscheck -l en connectivity -c "Oracle Client" -d "Oracle 9">c:\result.txt
```

Related Topics

- [Check tool—function overview](#) on page 18
- [Displaying help on the *cscheck* tool](#) on page 17

- [Check tool—find](#) on page 21
- [Check tool—accessdriver](#) on page 23
- [Check tool—middleware](#) on page 22
- [Check tool—ping](#) on page 25

Check tool—ping

This function attempts to access a given database using the supplied details.

Figure 2-10: *ping* syntax

cscheck ping|pgl-m{ Connection Server access mode }-**c**{ network layer }-**d**{ database client }-**u**{ user name }-**p**{ password }-**s**{ datasource }-**t**{ database }-**r**{ host name }-**j**{ PID }

Table 2-5: Function input parameters

Connection Server access mode (-m)	The mode in which the client application accesses Connection Server: <ul style="list-style-type: none"> • local: Connection Server is running on the local machine. • corba: Connection Server is running on a CORBA server. • http: Connection Server is running on a HTTP server. • java: Connection Server uses a Java data access driver on the local machine.
network layer (-c)	The database middleware for the connection to check, as returned by the <i>find</i> function.
database client (-d)	The database type, as returned by the <i>find</i> function.
user name (-u)	A valid user name for the database.

password (-p)	The password for the user name.
datasource (-s)	The server on which the database is running.
database (-t)	The database server.
host name (-r)	For CORBA mode, the computer hosting Connection Server.
PID (-i)	For CORBA mode, the process number of the Connection Server to ping through.

Example: Pinging an Oracle database

The following command checks access for:

- Connection Server access mode: **local**, that is, the database runs on the local machine.
- Network layer: **Oracle Client**
- Database: **Oracle 8.1**
- Datasource: **Harlaxton**
- User name: **efashion**
- Password: **X2345**

```
cscheck ping -m local -c "Oracle Client" -d "Oracle 8.1" -u  
"efashion" -p "X2345" -s "Harlaxton"
```

Example: Pinging a Sybase database using CORBA

The following command checks access for:

- Connection Server access mode: **CORBA**, that is, Connection Server runs on a CORBA server.
- Network layer: **Sybase**
- User name: **syadmin**

- Password: **password**
- Datasource: **Sybase Adaptive Server 15**
- Database: **SY1**
- Database host: **sybasehost**
- Process ID: **456**

```
cscheck ping -m corba -c "Sybase Open Client" -d syb15 -u  
"syadmin" -p "password" -s "Sybase Adaptive Server 15" -t  
"SY1" -r "sybasehost" -i 456
```

Related Topics

- [Check tool—function overview](#) on page 18
- [Check tool—find](#) on page 21

Accessing the New Connection wizard from the Designer Connections list

You can access the New Connection Wizard from any Business Objects application that allows you to create a new connection. For example, to access the New Connection wizard from Designer:

1. From the Designer user interface, select **Tools, Connections**.
2. Click **Add**.

The New Connection wizard **Welcome** page appears.

Related Topics

- [Accessing the New Connection wizard from the Parameters dialog box](#) on page 28

Accessing the New Connection wizard from the Parameters dialog box

You can access the New Connection Wizard from any Business Objects application that allows you to create a new connection. For example, to access the Connections Wizard from the Universe parameters dialog box:

1. Open Designer and select a Universe to load.
The Universe appears in the Designer work area.
2. From the **File** menu, select **Parameters**.
The **Universe Parameters** dialog box appears.
3. From the **Universe Parameters** dialog box, click the **New** button.
The New Connection wizard starts.

Related Topics

- [Accessing the New Connection wizard from the Designer Connections list](#) on page 27

Using the New Connection Wizard

The **New Connection** wizard steps you through the process of creating a connection. Use it to select a target data source, and configure the data access driver.

Use the New Connection wizard to set the following parameters:

Connection wizard stage	Description
Database middleware	Locate the database middleware to connect to, and assign a name for the connection.
Login parameters	Set the connection mode, and specify the login details for connecting to the middleware.

Connection wizard stage	Description
Configuration parameters	Configure connection parameters, for example the connection timeout details, array fetch and bind sizes, and login time-out. These vary depending on the database to which you are connecting.
Custom parameters	Customize parameters to optimize your connection. These vary depending on the database to which you are connecting.

Note: The parameters that the New Connection wizard displays vary depending on the options that you select, for example the database middleware, and the connection mode.

Related Topics

- [Database Middleware Selection dialog box](#) on page 30
- [Login Parameters dialog box](#) on page 32
- [Configuration parameters dialog box](#) on page 34
- [Custom Parameters dialog box](#) on page 35

Creating a new connection

You use the **New Connection** wizard to create a data access connection. The wizard displays a series of screens that you use to enter your connection configuration details.

The choices that are available in each screen can vary depending on the Business Objects application that you are using, and the database to which you are connecting.

- The **Database middleware selection** dialog box lists the available database vendors, databases, and middleware. Locate the database you want, and expand the database node to locate the driver that you want to use.

- Use the **Login parameters** dialog box to enter the login and connection information for the database. These details vary depending on the driver that you selected.

On this page you can also test the connection. If the test fails, check the credentials that you entered, or check with your database administrator.

- Use the **Configuration parameters** dialog box to configure the parameters that control the connection.
- Use the **Custom Parameters** dialog box to configure any specific custom parameters that apply to the connection. This page is available only for specific connections.

Values on the **Custom Parameters** dialog box should only be modified by an advanced user, database administrator, or Business Objects administrator. This page contains parameters that can be used to override the default connection settings. You would typically adjust these parameters to resolve performance issues.

- The **Connections List** dialog box lists the available connections. New connection that you define appear on this list . To test your connection, select the connection and click the **Test** button. A message appears, confirming that the connection is operating. If an error occurs, edit the connection to re-configure the parameters.

Related Topics

- [Database Middleware Selection dialog box](#) on page 30
- [Login Parameters dialog box](#) on page 32
- [Configuration parameters dialog box](#) on page 34
- [Custom Parameters dialog box](#) on page 35

Database Middleware Selection dialog box

The **New Connection** wizard's **Database Middleware Selection** dialog box contains the following settings:

- **Connection Type**: determines who can use the connection to access data. The options are explained in detail below.
- **Connection Name**: enter a name for the connection.

- **Filter Stored Procedures Network Layers:** select this check box to display only those datasources that support procedures stored in the database to deliver data. In order to use a JavaBean datasource, you must select this checkbox since a JavaBean datasource uses procedures stored in the JavaBean. If you do not select this checkbox, any installed JavaBean drivers do not appear in the selection list.
- A list of the available data access drivers: you expand list items and select the driver that you want.

You can create three types of connections with Designer:

- Personal
- Shared
- Secured

Personal connections

This connection type restricts data access to the universe creator, and the computer on which it was created. You can use personal connections to access personal data on a local machine only. You cannot use personal connections to distribute universes.

Connection parameters are stored in the PDAC.LSI file located in the LSI folder in the Business Objects 12.0 folder in your user profile directory, for example:

```
C:\Documents and Settings\\Application Data\Business Objects\Business Objects 12.0\lsi
```

These parameters are static and cannot be updated. Personal connections are unsecured when used with Business Objects products security.

Shared connections

This connection type allows access to data for all users. These connections are unsecured when used with Business Objects products security.

Connection parameters are stored in the SDAC.LSI file located in the LSI folder in the Business Objects 12.0 folder in your user profile directory, for example:

```
C:\Documents and Settings\<user name>\Application Data\Business  
Objects\Business Objects 12.0\lsi
```

Secured connections

This connection type centralizes and controls access to data. This is the safest type of connection, and should be used to control access to sensitive data.

You can create secured connections with Designer. You must use secured connections if you want to distribute universes through the CMS. Secured connection details are stored in the CMS.

Secured connections can be used and updated at any time.

Login Parameters dialog box

The **New Connection** wizard's **Login Parameters** dialog box can contain the following parameters:

Parameter	Description
Authentication mode	<ul style="list-style-type: none"> • Use specified user name and password: uses the login details as authentication. • Use Business Objects credential mapping: the user is prompted for a database user password associated with their BusinessObjects account to refresh a report. This is set using the parameters dbuser and dbpass. These are set at the administrative level. Refer to Business Objects Enterprise Administrator's Guide for information on setting up this option. • Use Single Sign On when refreshing reports at View Time: When selected, the username and password used to access the CMS are automatically used as the database login parameters. See the Business Objects Enterprise Administration guide for information on setting Single Sign-On (SSO).
User name	Your database user name. This is normally assigned to you by the database administrator.
Password	Your database password. This is normally assigned to you by the database administrator.
Datasource (<Host>:<port>):	The data source details.
Database	The database name.

Configuration parameters dialog box

The **Configuration Parameters** dialog box contains parameters that you can set to override default configuration options. These configuration parameters override:

- Any corresponding parameters set in the `cs.cfg` file
- Any corresponding parameters set in the `<driver>.sbo` file.

You can set the following parameters:

Connection pool mode	From the drop down list, select the method to use to keep the connection active. Only necessary if using a connection pool.
Pool timeout	If you select Keep the pool active for in the previous field, specifies the length of time to keep the connection open.
Array fetch size	<p>Enter the maximum number of rows authorized with each fetch.</p> <p>If you enter 20, and your query retrieves 100 rows, the connection executes 5 fetches to retrieve your data.</p> <p>If you enter 1, array fetch is deactivated and data is retrieved row by row.</p> <p>Note: Deactivating array fetch is the safest way of retrieving your data but row-by-row retrieval slows down server performance. The greater the value in the Array fetch size option, the faster your rows are retrieved. You must, however, ensure you have adequate client system memory.</p> <p>The default value is 20.</p>

Array bind size	Enter the size of the bind array that Connection Server uses before sending to the repository. The bind array is the area in memory where Connection Server stores a batch of data to be loaded (sent to the repository). When the bind array fills, it is transmitted to the database. Generally, the larger the bind array, the more rows (n) can be loaded in one operation, and the better your performance.
Login timeout	Specifies the number of seconds before a connection attempt times out and an error message is displayed.

Related Topics

- [SBO parameter categories](#) on page 56
- [About global parameters](#) on page 42

Custom Parameters dialog box

Use this dialog box to configure any custom configurations that apply to the connection. The parameters that appear depend on the other parameters that you have set. The parameters set here override:

- Any corresponding parameters set in the `cs.cfg` file
- Any corresponding parameters set in the `<driver>.sbo` file.

For information on the parameters, refer to the SBO file parameter reference information.

Related Topics

- [About global parameters](#) on page 42
- [SBO parameter categories](#) on page 56

About JDBC connections

A set of data access drivers are installed when you install your Business Objects application software. You can use these data access drivers to create connections to databases.

Business Objects software also includes configuration files for using JDBC drivers to access your databases. To use these drivers, you:

- Obtain the java driver software from your database supplier.
- Modify the supplied configuration files.

For an up-to-date list of supported JDBC drivers, check the Business Objects Support Site at www.businessobjects.com, or contact your BusinessObjects representative.

Creating a JDBC connection

In order to create a JDBC connection:

- Obtain the necessary JDBC driver software for the database, and copy the files to your system. The driver software consists typically of one or more jar files. Note the path details for these files.
- Ensure that you have the database access details to hand, for example the login and password details.

To create a JDBC connection, use the following procedure:

1. Check if there are any DLLs that the driver uses, and ensure that they are accessible by the system. For example, the SQLServer 2005 JDBC driver uses DLLs. The DLL's directory needs to be included in the **PATH** environment variable.
2. Navigate to the directory that contains the `jdbc.sbo` file. For example, on a Windows system, the configuration files are located in the following paths, where *connectionserver-install-dir* is the directory where the Connection Server software is installed: *connectionserver-install-dir* \connectionServer\jdbc
3. Use an XML editor to open the `jdbc.sbo` file for editing.
4. Add the required `.jar` file details to the **ClassPath** area. Include the fully qualified path names when specifying these files, for example:

```
<Path>C:\JDBC Drivers\MSSQLSERVER2000\ msutil.jar</Path>
```

Note: These files need to be installed on the machine running the Business Objects application.

Refer to the information on the JDBC sbo example file structure for details.

5. Locate the **Driver Capabilities** parameter, and check that it is set to either **Procedure** or **Queries**.

If it is not, the JDBC driver is unavailable from the **New Connection** wizard.

6. Save and close the .sbo file.
7. Run the Connection Wizard. The JDBC driver that you have configured appears in the list of available connections. Select the JDBC driver and use the Wizard and configure the connection .

When you complete this task, the connection is available for use.

Related Topics

- [JDBC SBO example file structure](#) on page 37
- [Before you create a connection](#) on page 16

JDBC SBO example file structure

This lists an example of the section of the JDBC SBO file that you need to modify. This SBO file is for Microsoft SQLServer 2000 and Microsoft SQLServer 2005.

```
<JDBCdriver>
  <ClassPath>
    <Path>C:\JDBC Drivers\MSSQLSERVER2000\
      msbase.jar</Path>
    <Path>C:\JDBC Drivers\MSSQLSERVER2000\
      msutil.jar</Path>
    <Path>C:\JDBC Drivers\MSSQLSERVER2000\
      mssqlserver.jar</Path>
  </ClassPath>
  <.....>
</JDBCdriver>
```

About JavaBean connections

Developers can create JavaBeans that you can use as data sources. You can create connections using these JavaBeans. Typically, these JavaBeans provide access to a datasource. In order to create a JavaBeans connection, the developers who create the JavaBean will supply:

- The required .jar files.
- Any other files that the JavaBean requires.
- Any specific configuration details that the JavaBeans driver requires.

Within a JavaBeans driver, data-retrieval procedures are configured as stored procedures. When configuring a JavaBeans connection, on the New Connection wizard's **Database Middleware Selection** screen, you must select the **Filter Stored Procedures Network Layers** check-box. If you do not, the New Connection wizard does not display the JavaBeans drivers that are available.

Related Topics

- [Database Middleware Selection dialog box](#) on page 30

Creating a JavaBean connection

To create a JavaBeans connection, use the following procedure:

1. Use an XML editor to open the `javabeans.sbo` file for editing. For example, on a Windows system, the configuration files are located in the following paths, where *connectionserver-install-dir* is the directory where the Connection Server software is installed: `connectionserver-install-dir\connectionServer\javabeans`
2. Add the required .jar file details to the **ClassPath** area. Include the fully qualified path names when specifying these files.

Note: These files need to be installed on the machine running the Business Objects application.

Refer to the information on the `JavaBean.sbo` example file structure for details.

3. Save and close the .sbo file.

4. Perform any other configuration tasks specified by the JavaBeans developer.
5. Run the Connection Wizard. The JavaBeans datasource that you have configured should appear in the list of available connections. Select the JavaBeans datasource and use the Wizard to configure the connection.

When you complete this task, the connection is available for use with a Business Objects application.

Related Topics

- [JavaBean SBO example file structure](#) on page 39
- [Database Middleware Selection dialog box](#) on page 30
- [PrimaryKey Available](#) on page 79

JavaBean SBO example file structure

This section contains an example of a JavaBeans SBO file.

```
<DataBase Active="Yes" Name="Excel
Spreadsheet">
  <JavaBean>
    <ClassPath>
      <Path>$ROOT$/beans/bean_excel.jar
    </Path>
    </ClassPath>
    <Parameter Name="JavaBean Class">com.
businessobjects.beans.excel.Excel
    </Parameter>
    <Parameter Name="URL Format">$DATASOURCE$
    </Parameter>
  </JavaBean>
  <Parameter Name="Family">Java Beans
</Parameter>
  <Parameter Name="Description File">
bean_excel</Parameter>
  <Parameter Name="Authentication Mode">
Bypass</Parameter>
  <Parameter Name="Extensions">bean_excel,
javabean</Parameter>
</DataBase>
</DataBases>
```

2 | Creating a connection

Creating a JavaBean connection



Configuring data access global parameters

About global parameters

You can configure the global parameter values that apply to all connections. You can do this to improve performance, or to resolve issues with the connection that arise.

Data access global parameters are maintained in the `cs.cfg` file. This is an XML file that contains Connection Server configuration parameters, and default configuration parameters that apply to all data access drivers.

To override these global settings, you can configure settings in each driver's `.sbo` file.

Related Topics

- [Configuring driver parameters](#) on page 52

About the `cs.cfg` global configuration file

On a Windows system, the `cs.cfg` file is stored in the following location:

- `connectionserver-install-dir\connectionServer`

In the `cs.cfg` file, you can configure parameters in the following sections only:

- **Settings**

This section defines Connection Server global configuration parameters, including client access types such as CORBA. These parameters can be overridden by corresponding settings in the data access driver configuration file: `<driver>.sbo`, where `<driver>` is the name of the data access driver to which the `.sbo` file relates.

- **DriverDefaults**

These parameters apply to all data access drivers.

- **Distribution**

You configure the settings in this section if you use CORBA.

- **Traces**

You can set trace parameters that allow the recording of connection activity through Connection Server in log files. Refer to the release notes for information on generating traces.

The remaining section, **Locales**, defines the operating system charset for each available language. The parameters in this section must not be modified.

Viewing and editing cs.cfg

You can view and edit parameters in `cs.cfg` as follows:

1. Browse to the directory that stores the `cs.cfg` file. For example, on a Windows system:

connectionserver-install-dir\connectionServer\cs.cfg where *connectionserver-install-dir* is the path where your Connection Server software is installed.
2. Open `cs.cfg` in an XML editor.
3. Expand sections as required.
4. Set parameters by either adding new parameters and values, or modifying existing parameter values.
5. Check that the document is valid against the DTD, then save and close the file.

Configuring the <driver defaults> parameters

The **<driver defaults>** section of the `cs.cfg` file contains the default values that apply to all data access drivers. These default values are overridden for a specific driver by corresponding values set in the `<driver>.sbo` file.

Refer to the SBO file parameter reference for information on the parameters that you can set.

Related Topics

- [Data access configuration files](#) on page 52
- [Configuring driver parameters](#) on page 52
- [SBO parameter categories](#) on page 56

Configuring <Settings> parameters

The <Settings> section of `cs.cfg` defines settings that apply to all drivers, and cannot be customized for individual data access drivers. Some settings can be defined either for **library** version or **server** version of Connection Server:

- **Library**: In this mode, Connection Server is included in the client process.
- **CORBA**: In this mode, Connection Server is a CORBA server and is accessed remotely. In this mode, Connection Server serves two different kinds of clients: HPPT and CORBA clients.

If your environment uses **CORBA**, you configure some parameters in the <Server> section of `cs.cfg`.

The <Settings> parameters are listed alphabetically. To view or edit parameters, open `cs.cfg` in an XML editor, and go to the `Settings` section, then `Parameters` section. In the file, each parameter is defined in the following tag:

```
<Parameter Name="parameter">value</Parameter>
```

where *parameter* is the name of the parameter, and *value* is the value to which the parameter is set.

Each parameter is shown with the following information:

- Example of how the parameter appears in the XML file. When the parameter is available in library and server mode, an example of each is shown.
- Description of the parameter
- Possible values that can be set for the parameter (where applicable)
- Default value for the parameter

Charset List Extension

```
<Parameter Name="CharSet List Extension">crs</Parameter>
```

Description	Note: Do not change this setting. Sets the file extension for character set files.
Default	crs

Config File Extension

<Parameter Name="Config File Extension">sbo</Parameter>

Description	Note: Do not change this setting. Sets the file extension for general configuration files.
Default	sbo

Description Extension

<Parameter Name="Description Extension">cod</Parameter>

Description	Note: Do not change this setting. Sets the file extension for the connection description files.
Default	cod

Enable Failed Load

<Parameter Name="Enable Failed Load">Yes</Parameter>

Description	Determines action taken when a driver fails to load. The parameter lets you choose whether you want a usable connection possibly without all drivers operating, or a fatal error and no functionality when a driver fails to load.
Values	Yes: Connection Server generates a fatal error when a driver fails to load. No: Connection Server generates a non-fatal error when a driver fails to load.
Default	Yes

Related Topics

- [Load Drivers On Startup](#) on page 46

Load Drivers On Startup

Load Drivers On startup applies to both **Library** and **Server** modes.

Library

```
<Settings>
<Parameter Name="Load Drivers On Startup">No</Parameter>
...
</Library/>
```

Server

```
<Server>
<Parameter Name="Load Drivers On Startup">Yes</Parameter>
...
</Server>
```

These are described below.

Description	Determines how driver libraries are loaded.
Values	<ul style="list-style-type: none"> • Yes: All installed drivers are loaded during the initialization phase. • No: Drivers are loaded on demand.

Default	Library mode: No Server mode: Yes
---------	--

Max Pool Time

Max Pool Time is available for **Library** mode and **Server** mode.

Library

```
<Settings>  
<Parameter Name="Max Pool Time">-1</Parameter>  
</Library/>
```

Server

```
<Server>  
  <Parameter Name="Max Pool Time">60</Parameter>  
</Server>
```

Description	Determines the maximum connection idle lifetime in the connection pool regardless of the value defined in the connection. You can set Max Pool Time for two types of Connection Server deployments: Library: Value applies to nodes that have Connection Server installed with desktop or other server products. Server: Value applies to Connection Server stand alone server installations. For information on deploying Connection Server on a dedicated node see the <i>Deployment guide</i> .
Default	Library mode: -1 Server: 60
Values	-1: No timeout, keep alive for the whole session. 0: Connection not managed by the pool. >0: Idle lifetime (in minutes).

SQL External Extension

```
<Parameter Name="SQL External Extension">rss</Parameter>
```

Description	Note: Do not change this setting. Sets the file extension for external SQL files.
Default	rss

SQL Parameter Extension

```
<Parameter Name="SQL Parameter Extension">prm</Parameter>
```

Description	Note: Do not change this setting. Sets the file extension for SQL parameter files.
Default	prm

Strategies Extension

```
<Parameter Name="Strategies Extension">stg</Parameter>
```

Description	Note: Do not change this setting. Sets the extension for strategy files.
Default	stg

Configuring CORBA access

Connection Server can operate in the following two modes:

- **Library:** In this mode, Connection Server is included in the client process.
- **CORBA:** In this mode, Connection Server is a CORBA server and is accessed remotely. In this mode, Connection Server serves two different kinds of clients: HTTP and CORBA clients.

In `cs.cfg`, parameters in the `<Settings>` section control the access method.

In the `<Settings>` section:

- Parameters in the `<Library>` section control Library mode. Most Business Objects products use Connection Server in Library mode.
- Parameters in the `<Server>` section control CORBA access.

Parameters defined in `<Library>` or `<Server>` take precedence over any parameter definitions set in other parts of the `<Settings>` section. For example, with CORBA, parameters defined in the `<Server>` section can override parameters set in other areas for applications that use `<Library>` access.

The settings that apply to CORBA access are:

- Load Drivers On Startup
- Max Pool Time

In addition to these settings, you must configure settings in the Distribution section.

Related Topics

- [Configuring the Distribution section for CORBA access](#) on page 50
- [Load Drivers On Startup](#) on page 46
- [Max Pool Time](#) on page 47

Configuring the Distribution section for CORBA access

You set the following parameters in the `<Distribution>` section of `cs.cfg` when using Connection Server with CORBA.

The `<Protocols>` section contains the default values that Connection Server uses to process requests coming from CORBA clients or HTTP clients. For CORBA access, configure the section as follows:

```
<Protocol Name="CORBA" Active="Yes"/>  
<Protocol Name="HTTP" Active="No"/>
```

The `<Lookup>` section contains parameters for internal use, and must not be modified.



Configuring data access driver parameters

4

chapter



Configuring driver parameters

To configure data access for a particular data access driver, you can edit the driver's XML parameter files to adjust the parameter settings.

Note: For each Business Objects application that uses Connection Server, the associated Readme file contains information on command line utilities that you can use to check your RDBMS and data access driver configuration. These utilities can create log files that trace Web Intelligence server activity. Refer to the readme for the release for instructions on the use of these utilities.

Related Topics

- [About global parameters](#) on page 42

Data access configuration files

The following configuration files control data access driver configurations for each defined connection:

- The `cs.cfg` file defines global parameters that apply to all connections.
- There is a separate configuration file for each data access driver. These files are named `<driver>.sbo`, where `<driver>` is the database network layer to which the configuration file applies.

The parameters set in the `Settings` section of `cs.cfg` are overridden by corresponding settings in the data access driver configuration files: `<driver>.sbo`

The configuration files are located in the following paths, where *connection server-install-dir* is the directory where the Connection Server software is installed.

- The `cs.cfg` file is in the following directory:
`connectionserver-install-dir\connectionServer\`
- Each `.sbo` file is in a sub-directory of this directory, where the sub-directory is named after the database network layer, for example for Oracle databases:

`connectionserver-install-dir\connectionServer\oracle`

Related Topics

- [About global parameters](#) on page 42
- [Installed SBO files](#) on page 53
- [Configuring the <driver defaults> parameters](#) on page 43

Installed SBO files

The following <driver>.sbo files are installed by default.

For an up-to-date list of supported drivers, check the Business Objects Support Site at www.businessobjects.com, or contact your BusinessObjects representative.

Data access driver	SBO file
Essebase	essebase.sbo
IBM DB2	db2.sbo iseries.sbo
Informix	informix.sbo
JDBC provides JDBC drivers for databases. Check the Business Objects support website, or the jdbc.sbo file for details of the databases supported for JDBC connections.	jdbc.sbo
Microsoft SQL Server Microsoft Analysis Services	odbc.sbo oledb_olap.sbo
MySQL	odbc.sbo
ODBC	odbc.sbo
Oracle	oracle.sbo
Red Brick	odbc.sbo

Data access driver	SBO file
SAP	sap.sbo
Sybase	sybase.sbo
Teradata	teradata.sbo

Editing an SBO file

You can open an SBO file for viewing or editing as follows:

Note: Before opening an SBO file, make a backup copy of the file. Some configuration parameters must not be edited. If you change or delete them it could affect the operation of your Business Objects applications.

1. Browse to the directory that stores the SBO file for your target data access driver. SBO files are stored in the following location, where <driver> is the name of the database software that the connection uses:

```
\\<INSTALDIR>\win32_x86\dataAccess\connectionServer\\<driver>.sbo
```

2. Open the <driver>.sbo file in a XML editor.
3. Expand sections as required.
4. Locate the appropriate tag for the value to change, and change the value. Parameters appear in the format: <Parameter Name="parameter">value</Parameter> where *parameter* is the name of the parameter, and *value* is the value attributed to the parameter.
5. Check that the file is valid against the DTD, save and close the file.



SBO file parameter reference

SBO parameter categories

The configuration parameters in this guide are listed in the following categories. Each category represents a database that has a separate `sbo` file.

- Defaults
- Informix
- JavaBean
- JDBC
- ODBC
- ODBC3
- OLE DB
- Sybase ASE/CTLIB

Each parameter is shown with the following information:

- Example of how the parameter appears in the XML file
- Description of the parameter
- Possible values that can be set for the parameter
- Default value for the parameter

SBO file structure

There is a `<driver>.sbo` file for each supported database network layer. Each `<driver>.sbo` file is divided into the following sections:

File section	Description
Defaults	This section contains the default configuration parameters that apply to all database middleware that uses the data access driver. These parameters override any corresponding values set in the database middle-ware.

File section	Description
Databases	<p>This section contains a sub-section for each database middleware that is supported by the data access driver. The <code>Active</code> parameter specifies if middleware support is activated or not.</p> <p>Values are YES or NO.</p> <p>Each middleware section can contain the following parameters:</p> <ul style="list-style-type: none">• Name: Names of the middleware supported by the data access driver. The middleware name values set here appear in the Database Middleware page of the new connection wizard.• Aliases: Names of older middleware versions no longer officially supported by the data access driver, but that are still in use. You can add an alias parameter for an older middleware version so that existing connections use the current data access driver instead. You can set configuration parameters specific to the old middleware as parameters of the new alias. You can create new connections using the alias.• Parameters: Configuration parameters with values that apply specifically to a middleware. Values set for parameters listed here override the values set for the same parameters in the Defaults section.

Default SBO parameters

These SBO parameters are defined in `cs.cfg`, or under the `Defaults` section of the SBO file.

Related Topics

- [SBO parameter categories](#) on page 56

Array Bind Available

```
<Parameter Name="Array Bind Available">True</Parameter>
```

Description	Specifies whether or not the database supports Array Binds.
Values	True: the database supports Array Binds. False: the database does not support Array Binds.
Default	False

Array Bind Size

```
<Parameter Name="Array Bind Size">5</Parameter>
```

Description	Specifies the number of rows exported with each INSERT command.
Values	An integer that specifies the number of rows that are exported with each INSERT .
Default	The value set in the <code>cs.cfg</code> file.

Array Fetch Available

```
<Parameter Name="Array Fetch Available">True</Parameter>
```

Description	Specifies whether or not the Array Fetch method is supported.
Values	True: the Array Fetch method is supported. False: the Array Fetch method is not supported.
Default	The value set in the <code>cs.cfg</code> file.

Array Fetch Size

```
<Parameter Name="Array Fetch Size">10</Parameter>
```

Description	<p>Specifies the number of rows of data retrieved with each Array Fetch method. The optimum number depends on your system's performance:</p> <ul style="list-style-type: none">• If the number is low, the system retrieves small amounts of data many times. This can affect performance.• If the number is high, the system performs fewer retrieval operations, but it requires more memory for each.
Values	<p>An integer that specifies the number of rows that are retrieved with each Array Fetch.</p> <p>1 : specifies that Array Fetch is deactivated.</p>
Default	Value set in the <code>cs.cfg</code> file.

Catalog Name Max Size

```
<Parameter Name="Catalog Name Max Size">1024</Parameter>
```

Description	Specifies the maximum length in characters of a database catalog name.
Values	An integer that specifies the maximum database catalog name length in characters.
Default	The value set in the database middleware.

Catalog Separator

```
<Parameter Name="Catalog Separator">-</Parameter>
```

Description	Specifies the separator character that is used between identifiers.
Values	The separation character to use.
Default	If not specified, Connection Server uses the separator specified in the database middleware.

CharSet

```
<Parameter Name="CharSet">UTF8</Parameter>
```

Description	Specifies the character set of the data returned by the database middleware.
Values	UTF8: 8-bit UCS/Unicode Transformation Format UCS2: 2-byte Universal Character Set
Default	If not specified, Connection Server uses the separator specified in the database middleware.

Column Name Max Size

```
<Parameter Name="Column Name Max Size">1024</Parameter>
```

Description	Specifies the maximum length in characters of a database column name.
Values	An integer that specifies the maximum database column name length in characters.

Default	The value set in the database middleware.
---------	---

Connection Shareable

```
<Parameter Name="Connection Shareable">False</Parameter>
```

Description	Specifies if the connection configuration is shareable between different connections. Operates in conjunction with the <code>Shared Connection</code> parameter.
Values	True: the connection configuration can be shared between connections. False: the connection configuration cannot be shared between connections
Default	False

Related Topics

- [Shared Connection](#) on page 70

Cost Estimate Available

```
<Parameter Name="Cost Estimate Available">False</Parameter>
```

Description	Specifies if the database middleware supports cost estimation.
Values	True: the middleware supports cost estimation. False: the middleware does not support cost estimation.
Default	False

Description File

```
<Parameter Name="Description File">oracle</Parameter>
```

Description	<p>Note: Do not edit this parameter.</p> <p>Specifies the name of the file that holds the connection wizard input field labels.</p>
-------------	--

Driver Capabilities

```
<Parameter Name="Driver Capabilities">Procedures , Query</Parameter>
```

Description	<p>The capabilities of the driver, that is whether it can access stored procedures and queries available in the database software. This parameter is set typically using the New Connection wizard. You can include both values in the parameter.</p> <p>Note: This parameter must be set to Procedures for a JavaBeans driver. The functionality of a JavaBeans driver is defined as stored procedures as far as Business Objects applications are concerned.</p>
Values	<p>Procedures: the driver can access the data retrieval procedures that are defined in the database software.</p> <p>Query: the driver can access the data retrieval queries that are defined in the database software.</p>
Default	Procedures

Escape Character

```
<Parameter Name="Escape Character">-</Parameter>
```

Description	Specifies the character to use to escape strings of special characters, for example patterns.
Values	The character to use as the escape character.
Default	If not specified, Connection Server retrieves the value from the middleware.

Extensions

```
<Parameter Name="Extensions"></Parameter>
```

Description	Note: Do not modify the settings in this parameter.
-------------	--

Family

```
<Parameter Name="Family">Sybase</Parameter>
```

Description	Note: Do not edit this parameter. Specifies the family of the database engine that is displayed in the Database Middleware Selection page of the New Connection wizard. The set of middleware that corresponds to your license is displayed on this page in a tree view.
-------------	--

Field Size Factor

```
<Parameter Name="Field Size Factor ">1</Parameter>
```

Description	<p>Note: Do not change this value.</p> <p>The value to use to compute the size of returned content expressed as field characters when the content is returned as bytes. This is used for DB2 only.</p>
-------------	---

Force Execute

<Parameter Name="Force Execute">Never</Parameter>

Description	<p>Specifies whether or not the SQL query is executed before retrieving results. Supported by the following:</p> <ul style="list-style-type: none"> • ODBC • OLE • DB • JDBC
Values	<p>Never: The SQL query is never executed before retrieving results.</p> <p>Procedures: Execute only for stored procedures.</p> <p>Always: The SQL query is always executed before retrieving the results.</p>
Default	Never

Identifier Case

<Parameter Name="Identifier Case">LowerCase</Parameter>

Description	<p>Specifies how the database handles the case behavior of simple identifiers.</p>
-------------	--

Values	LowerCase : identifiers must be in lower case. UpperCase : identifiers must be in upper case. MixedCase : identifiers can be in mixed case. SensitiveCase : identifiers are case sensitive.
Default	If this setting is not specified, Connection Server retrieves the information from the database middleware.

Identifier Quote String

```
<Parameter Name="Identifier Quote String">-</Parameter>
```

Description	Specifies the character used to quote identifiers.
Values	The character used to quote identifiers.
Default	If this setting is not specified, Connection Server retrieves the information from the database middleware.

LIKE Escape Clause

```
<Parameter Name="Identifier Case">True</Parameter>
```

Description	Specifies if escape character use is supported in the LIKE clause.
Values	True : escape character use is supported in the LIKE clause. False : escape character use is not supported in the LIKE clause.
Default	If this setting is not specified, Connection Server retrieves the information from the database middleware.

Locale

```
<Parameter Name="Locale">en_us</Parameter>
```

Description	Specifies the locale of the middleware.
Values	<p>The locale of the middleware.</p> <ul style="list-style-type: none"> Language is specified in ISO-639 standard Country is specified in ISO-3166 standard <p>For example: en_US for English United States.</p>
Default	If this setting is not specified, Connection Server retrieves the information from the database middleware.

Max Rows Available

```
<Parameter Name="Max Rows Available">True</Parameter>
```

Description	Specifies if the driver supports the Max Rows function to limit the maximum number of rows that can be retrieved from a datasource.
Values	<p>True: The driver supports the Max Rows function .</p> <p>False: The driver does not support the Max Rows function.</p>
Default	False

Optimize Execute

```
<Parameter Name="Optimize Execute">False</Parameter>
```

Description	Specifies whether or not Connection Server optimizes the execution of SQL queries. This parameter is supported by Oracle and ODBC drivers only.
Values	True: specifies that where possible, SQL queries are optimized on execution. False: SQL queries are not optimized for execution.
Default	False

Owners Available

<Parameter Name="Owners Available">True</Parameter>

Description	Specifies whether or not Owners are supported by the target database.
Values	True: Specifies that owners are supported by the target database. False: Specifies that owners are not supported by the target database.
Default	Not specified: value retrieved from the database middle-ware.

Password_Encryption

<Parameter Name="Password Encryption">True</Parameter>

Description	Specifies whether or not to use the encryption password mechanism specified in the middleware for the password entered in the Connection details dialog box. This parameter is used only with Sybase. It is included in the Defaults section for future compatibility.
-------------	---

Values	<p>True: Specifies that the encryption password mechanism of the middleware is used.</p> <p>False: Specifies that the encryption password mechanism of the middleware is not used.</p>
Default	True

Procedure Name Max Size

```
<Parameter Name="Procedure Name Max Size">1024</Parameter>
```

Description	Specifies the maximum length in characters of a database procedure name.
Values	An integer that specifies the maximum database procedure name length in characters.
Default	The value set in the database middleware.

Procedure Parameter Name Max Size

```
<Parameter Name="Procedure Parameter Name Max Size">1024</Parameter>
```

Description	Specifies the maximum length in characters of a database procedure parameter name.
Values	An integer that specifies the maximum database procedure parameter name length in characters.
Default	The value set in the database middleware.

Qualifiers Available

```
<Parameter Name="Qualifiers Available">True</Parameter>
```

Description	Specifies whether or not Qualifiers are supported.
Values	True: Specifies that Qualifiers are supported. False: Specifies that Qualifiers are not supported.
Default	Not specified. Connection Server retrieves this information from the database middleware.

Query TimeOut Available

```
<Parameter Name="Query TimeOut Available">True</Parameter>
```

Description	Specifies whether or not Query TimeOut is supported by the database middleware, that is if a query that is running can be cancelled after a time period has expired.
Values	True: Specifies that Query TimeOut is supported by the database middleware . False: Specifies that Query TimeOut is not supported by the database middleware .
Default	False

Quoted Identifier Case

```
<Parameter Name="Quoted Identifier Case Available">True</Parameter>
```

Description	Specifies how the database treats the case of quoted identifiers.
-------------	---

Values	LowerCase: quoted identifiers must be in lower case. UpperCase: quoted identifiers must be in upper case. MixedCase: quoted identifiers can be in mixed case. SensitiveCase: quoted identifiers are case sensitive.
Default	Not specified. Connection Server retrieves the information from the middleware.

Schema Name Max Size

```
<Parameter Name="Schema Name Max Size">1024</Parameter>
```

Description	Specifies the maximum length in characters of the database schema name.
Values	An integer that specifies the maximum database schema name length in characters.
Default	The value set in the database middleware.

Shared Connection

```
<Parameter Name="Shared Connection">True</Parameter>
```

Description	Specifies whether or not the connection can be shared between different workflows. Operates in conjunction with the <code>Connection Shareable</code> parameter.
Values	True: Connections are shared if possible. False: Connections are not shared.
Default	False

Related Topics

- [Connection Shareable](#) on page 61

SQL External File

<Parameter Name="SQL External File">filename</Parameter>

Description	<p>Note: Do not edit this value.</p> <p>The SQL External file file holds configuration details used by the data access layer.</p>
-------------	--

SQL Parameter File

<Parameter Name="SQL Parameter File">oracle</Parameter>

Description	<p>The name of the file that stores database parameters. The extension of this file is: .prm</p> <p>You must ensure that this file is located in the same directory as the data configuration file (.sbo file).</p>
Values	<p>db2iseries for IBM DB2 iSeries</p> <p>db2udb for IBM DB2</p> <p>informix for Informix</p> <p>sqlsrv for Microsoft SQL Server 2000</p> <p>oracle for Oracle</p> <p>redbrick for Red Brick</p> <p>sybase for Sybase</p> <p>asiq for Sybase ASIQ</p> <p>teradata for Teradata</p>
Default	<p>The listed values.</p>

SSO Available

```
<Parameter Name="SSO Available">False</Parameter>
```

Description	Specifies whether or not Single Sign On is supported.
Values	True: Single Sign On is supported. False: Single Sign On is not supported .
Default	False

Strategies File

```
<Parameter Name="Strategies File">oracle</Parameter>
```

Description	Specifies the name, with no extension, of the Strategy file (.stg). This file contains the external strategies that Designer uses for automatic universe creation. Strategy files are stored in the same directory as the .sbo file.
Values	db2 for DB2 data access drivers. informix for Informix. oracle for Oracle. sybase for Sybase. teradata for Teradata.
Default	See values above.

Table Name Max Size

```
<Parameter Name="Table Name Max Size">1024</Parameter>
```

Description	Specifies the maximum length in characters of a database table name.
-------------	--

Values	An integer that specifies the maximum database table name length in characters.
Default	The value set in the database middleware.

Transactional Available

```
<Parameter Name="Transactional Available">Yes</Parameter>
```

Description	Specifies if SQL operations run against the database are run as block transactions or individually. This parameter is not listed by default in the .sbo file. Add it to the .sbo file if your data access driver does not support transactional mode.
Values	Yes: operations against the database are run as a block when committed. No: each SQL statement is immediately committed. That is, Autocommit is de-activated. Note: Do not use a driver with <code>Transactional Available=No</code> to access the Business Objects repository.
Default	Yes. This is set in the <code>cs.cfg</code> file.

Transaction Mode

```
<Parameter Name="Transaction Mode">AutoCommit</Parameter>
```

Description	Specifies the transaction mode that the database uses.
-------------	--

Values	AutoCommit: statements are committed automatically on completion of a request. Transactional: on completion, an explicit call either completes or rolls back the statement.
Default	If not specified, the value is retrieved from the database middleware.

Type

```
<Parameter Name="Type">Relational</Parameter>
```

Description	Specifies the Business Objects data source type. Note: This parameter must not be modified.
-------------	---

Unicode

```
<Parameter Name="Unicode">CharSet</Parameter>
```

Description	Specifies if the access driver can benefit from the Unicode configuration of the client middleware. This parameter appears as a driver default in the <code>cs.cfg</code> file. Its value applies to all data access drivers. It is not listed by default in the SBO file. If you want to over-ride the default value you add it to the Defaults section of the SBO file for the target data access driver.
Values	UTF8: 8-bit UCS/Unicode Transformation Format coding. CharSet: Character Set coding. UCS2: 2-byte Universal Character Set coding
Default	The value set in the <code>cs.cfg</code> file.

Version

```
<Parameter Name="Version">Relational</Parameter>
```

Description	Specifies the database version. Note: This parameter must not be modified.
-------------	--

XML Max Size

```
<Parameter Name="XML Max Size">2048</Parameter>
```

Description	Specifies the maximum size allowed for XML data.
Values	The maximum allowed XML size, in bytes.
Default	This varies depending on the database.

Informix SBO parameters

These parameters apply to the Informix SBO file. They are used to define the connection to an Informix database.

These parameters are defined in the `\\<INSTALDIR>\win32_x86\dataAccess\connectionServer\informix\informix.sbo` file.

Note: In addition to the SBO parameters, for Informix, in a Unix environment, you must modify the ODBC.INI file.

Related Topics

- [V5toV6DriverName](#) on page 76
- [Modifying the Informix ODBC.INI file](#) on page 76

Modifying the Informix ODBC.INI file

In order to use the data access layer with an Informix database in a **UNIX** environment, you need to modify the unicode configuration. This configuration information can be in either of the following locations:

- The `.odbc.ini` file located in your home directory.
 - The file specified by the **ODBCINI** environment variable.
1. Locate the `ODBC.INI` file or the file specified by the **ODBCINI** environment variable, and open it in a text editor.
 2. In the file, locate the **[ODBC]** section.
 3. Add the following line to the **[ODBC]** section:
UNICODE=UTF-8
 4. Close the file and save it.

Example: ODBC section with the configuration added

This code is an example of the **ODBC** section with the correct information added:

```
[ODBC]
UNICODE=UTF-8
```

V5toV6DriverName

```
<Parameter Name="V5toV6DriverName">{Informix 3.34 32 BIT}</Parameter>
```

Description	Specifies the conversion rule from Informix Connect to Informix ODBC . The value of this parameter determines which Informix Driver is used to define the ODBC Data Source Name (DSN) without the connection string..
Values	The exact name of the Informix driver installed on the machine.
Default	The value set in the <code>cs.cfg</code> file.

JavaBean SBO parameters

These parameters apply to the JavaBean SBO file. They are used to define a JavaBean connection.

These parameters are defined in the `\\<INSTALDIR>\win32_x86\dataAccess\connectionServer\javaean\javaean.sbo` file.

JavaBean Class

```
<Parameter Name="JavaBean Class">string</Parameter>
```

Description	Defines the entry point of the JavaBean that the Business Objects application uses. The entry point is the definition of a java class extending from the Bean interface specified through the com.businessobjects package.
Values	A fully-qualified class JavaBean class name.
Default	None.

URL Format

```
<Parameter Name="URL Format ">string</Parameter>
```

Description	<p>Specifies the URL Format.</p> <p>The JDBC specification does not specify the format of the connection string that it requires. Vendors use different kinds of URL format, for example:</p> <ul style="list-style-type: none">MySQL vendor : <code>jdbc:mysql://\$DATASOURCE\$/\$DATABASE\$</code>Oracle vendor : <code>jdbc:oracle:thin:@\$DATASOURCE\$: \$DATABASE\$</code>
-------------	--

Values	The URL Format
Default	None

JDBC SBO parameters

These parameters apply to the JDBC SBO file. They are used to define a JDBC connection.

These parameters are defined in the \\<INSTALDIR>\win32_x86\dataAccess\connectionServer\jdbc\jdbc.sbo file.

ForeignKeys Available

```
<Parameter Name="ForeignKeys Available">True</Parameter>
```

Description	Specifies if ForeignKeys can be retrieved.
Values	True: ForeignKeys can be retrieved. False: ForeignKeys cannot be retrieved.
Default	True

JDBC Class

```
<Parameter Name="JDBC Class">string</Parameter>
```

Description	The JDBC driver's fully qualified Java class.
-------------	---

Values	Depends on the vendor/datasource, for example : <ul style="list-style-type: none">• oracle.jdbc: OracleDriver for Oracle• com.ibm.db2.jcc.DB2Driver: for DB2
Default	None.

PrimaryKey Available

```
<Parameter Name="PrimaryKey Available">True</Parameter>
```

Description	Specifies whether or not the primary keys can be retrieved.
Values	True : Primary keys can be retrieved. False : Primary keys cannot be retrieved.
Default	True

ODBC SBO parameters

These parameters apply to the ODBC SBO file. They are used to define an ODBC SBO connection.

These parameters are defined in the \\<INSTALDIR>\win32_x86\dataAccess\connectionServer\odbc\odbc.sbo file.

Empty String

```
<Parameter Name="Empty String">NullString</Parameter>
```

Description	Specifies that certain functions, for example SQL tables, receive either an empty string or a null pointer to replace missing parameters.
-------------	---

Values	NullString : Specifies that a null string is used. EmptyString : Specifies that an empty string is used.
Default	EmptyString

Force Close Statement

<Parameter Name="Force Close Statement">True</Parameter>

Description	Specifies that the SQL statement is closed after it completes execution.
Values	True : the SQL statement is closed after it executes. False : the SQL statement is not closed after it executes.

ODBC Cursors

<Parameter Name="ODBC Cursors">No</Parameter>

Description	Specifies if the ODBC cursor library is used by the data access driver.
Values	Yes : the ODBC cursor library is used by the data access driver. No : the ODBC cursor library is not used by the data access driver.
Default	The value set in the <code>cs.cfg</code> file.

SQLDescribeParam Available

<Parameter Name="SQLDescribeParam Available">True</Parameter>

Description	Specifies whether or not the SQLDescribeParam Available mechanism is available.
Values	True: Specifies that the SQLDescribeParam Available mechanism is available. False: Specifies that the SQLDescribeParam Available mechanism is not available.
Default	The value set in the database middleware.

SQLMoreResults Available

```
<Parameter Name="SQLMoreResults Available">True</Parameter>
```

Description	Specifies whether or not the SQLMoreResults Available mechanism is supported.
Values	True: Specifies that the SQLMoreResults Available mechanism is supported. False: Specifies that the SQLMoreResults Available Available mechanism is not supported.
Default	The value set in the middleware.

ODBC3 SBO parameters

These parameters apply to the ODBC3 SBO file. They are used to define an ODBC3 connection.

These parameters are defined in the `\\<INSTALDIR>\win32_x86\dataAccess\connectionServer\odbc3\odbc3sbo\odbc3.sbo` file.

Connection Status Available

```
<Parameter Name="Connection Status Available">True</Parameter>
```

Description	Specifies whether or not the middleware can detect a bad connection.
Values	True: The middleware can detect a bad connection. False: The middleware cannot detect a bad connection.
Default	The value set in the middleware.

Native Int64 Available

```
<Parameter Name="Native Int64 Available">False</Parameter>
```

Description	Indicates if 64 bit integers can be handled directly by middleware.
Values	True: Specifies that 64-bit integers can be handled by the middleware. False: The Business Objects Data Access Layer emulates the Int64 methods.
Default	True

OLE DB SBO parameters

These parameters apply to the OLE DB SBO file. They are used to define an OLE DB SBO connection.

These parameters are defined in the \\<INSTALDIR>\win32_x86\dataAccess\connectionServer\oledb\oledb.sbo file.

Provider CLSID

```
<Parameter Name="Provider CLSID">MSDASQL</Parameter>
```

Description	Note: Do not edit this parameter. Specifies name of OLEDB provider. This parameter is used with OLEDB only.
-------------	---

Sybase ASE/CTLIB

These parameters apply to the Sybase ASE/CTLIB SBO file. These parameters are used to define a Sybase ASE/CTLIB connection.

These parameters are defined in the \\<INSTALDIR>\win32_x86\dataAccess\connectionServer\sybase\sybase.sbo file.

Quoted Identifier

<Parameter Name="Quoted Identifier">True</Parameter>

Description	Specifies whether or not quoted identifiers are supported.
Values	True: quoted identifiers are supported. False: quoted identifiers are not supported.
Default	The value set in the middleware.



Configuring SQL generation parameters for a universe



6

chapter



About SQL generation parameters for a universe

SQL generation parameters control the SQL queries that Business Objects applications generate to retrieve the contents of a Universe.

You can set these SQL generation parameters to operate at :

- The Universe level: you set these parameters when you create or modify a Universe.
- The database level: you set these parameters in the PRM file for the database. These parameters are overridden by any corresponding Universe-level PRM file settings.

Editing SQL generation parameters in a universe

Many of the parameters common to most supported RDBMS middleware are available for editing in the **Parameters** tab in the **Universe parameters** dialog box (**File > Parameters > Parameter**).

These parameters apply only to the active universe, and are saved in the UNV file. When you modify an SQL parameter for a universe in Designer, the value defined in Designer is used, and not the value defined in the PRM file associated with the data access driver for the connection.

You can modify the values for SQL parameters that determine SQL generation in products using the universe.

To edit SQL generation parameters in Designer:

1. From the **File** menu, select **Parameters**.
The **Universe Parameters** dialog box appears.
2. Click the **Parameter** tab.
The Parameter page appears.
3. Edit, add, or remove parameters as follows:

To...	Then do the following
Add a new parameter	<ul style="list-style-type: none"> Click any parameter in the list. Type a name in the Name box Type a value in the Value box. Click Add. <p>The new value appears at the bottom of the list</p>
Change name or value	<ul style="list-style-type: none"> Click a parameter in the list. Type a new name in the Name box Type a new value in the Value box. Click Replace. <p>The value is replaced by the new definition.</p>
Delete a parameter	<ul style="list-style-type: none"> Click the parameter that you want to remove from the list. Click Delete.

4. Click **OK**.

Note: The SQL generation parameter values that you set in a universe are only available to products using that universe.

Universe SQL parameters reference

This section provides an alphabetical reference for the SQL generation parameters listed in the Parameter page of the Universe Parameters dialog box in Designer. These are SQL parameters that are common to most data access drivers. Each parameter is valid for the universe in which it is set. Other RDBMS specific and connection parameters are listed in the data

access parameter (PRM) file for the target data access driver. Refer to the Data Access guide for a reference to the parameters in the PRM file.

ANSI92

ANSI92 = Yes|No

Values	Yes No
Default	No
Description	<p>Specifies whether the SQL generated complies to the ANSI92 standard.</p> <p>Yes: Enables the SQL generation compliant to ANSI92 standard.</p> <p>No: SQL generation behaves according to the PRM parameter OUTER_JOIN_GENERATION.</p>

AUTO_UPDATE_QUERY

AUTO_UPDATE_QUERY = Yes|No

Values	Yes No
Default	Yes
Description	<p>Determines what happens when an object in a query is not available to a user profile.</p> <p>Yes: Query is updated and the object is removed from the query.</p> <p>No: Object is kept in the query.</p>

BLOB_COMPARISON

BLOB_COMPARISON = Yes|No

Values	Yes No
Default	No
Can be edited?	No
Description	<p>Species if a query can be generated with a DISTINCT statement when a BLOB file is used in the SELECT statement. It is related to the setting No Duplicate Row in the query properties.</p> <p>Yes: The DISTINCT statement can be used within the query.</p> <p>No: The DISTINCT statement cannot be used within the query even if the query setting No Duplicate Row is on.</p>

BOUNDARY_WEIGHT_TABLE

BOUNDARY_WEIGHT_TABLE = Integer 32bits [0-9]

Values	Integer 32bits [0-9]
Default	-1
Description	<p>Allows you to optimize the FROM clause when tables have many rows.</p> <p>If the table size is greater than the entered value, the table is declared as a subquery:</p> <pre>FROM (SELECT col1, col2,....., coln, ,...., FROM Table_Name WHERE simple condition).</pre> <p>A simple condition is defined as not having a subquery, and not having EXCEPT or BOTH operators.</p>

Limitations	<p>Optimization is not implemented when:</p> <ul style="list-style-type: none"> • the operator OR is in the query condition • only one table is involved in the SQL • the query contains an outer join • no condition is defined on the table that is being optimized • the table being optimized is a derived table.
-------------	---

COLUMNS_SORT

COLUMNS_SORT = Yes|No

Values	Yes/No
Default	No
Description	<p>Determines the order that columns are displayed in tables in the Structure pane.</p> <p>Yes: Columns are displayed in alphabetical order</p> <p>No: Columns are displayed in the order they were retrieved from the database</p>

COMBINE_WITHOUT_PARENTHESIS

COMBINE_WITHOUT_PARENTHESIS = No

Values	Yes/No
Default	No

Description	<p>Specifies whether or not to encapsulate a query with parentheses when it contains UNION, INTERSECT or MINUS operators. Used with RedBrick.</p> <p>Yes Removes the parentheses.</p> <p>No Leaves the parentheses.</p>
-------------	---

COMBINED_WITH_SYNCHRO

COMBINED_WITH_SYNCHRO = Yes|No

Values	Yes No
Default	No
Description	<p>Specifies whether to allow a query to execute that contains UNION, INTERSECTION, or EXCEPT operators, and whose objects in each subquery are incompatible.</p> <p>Yes: Specifies that you do allow a query to execute that contains UNION, INTERSECTION and EXCEPT operators, and whose objects in each subquery are incompatible. This type of query generates synchronization (two blocks in the report).</p> <p>No: Specifies that you do not allow a query to execute that contains UNION, INTERSECTION and EXCEPT operators, and whose objects in each subquery are incompatible. When the query is executed the following error message is displayed: "This query is too complex. One of the subqueries contains incompatible objects." This is the default value.</p>

COMPARE_CONTEXTS_WITH_JOINS

COMPARE_CONTEXTS_WITH_JOINS = Yes|No

Values	Yes No
Default	No
Description	<p>Specifies how contexts are compared.</p> <p>Yes: The system verifies that the contexts give the same joins.</p> <p>No: The system verifies that the contexts give the same sets of tables. This is the default value.</p>

CORE_ORDER_PRIORITY

CORE_ORDER_PRIORITY = Yes|No

Values	Yes No
Default	No
Description	<p>Specifies in which order you want classes and objects to be organized once two or more universes are linked in Designer.</p> <p>Yes: Specifies that classes and objects follow the order defined in the kernel universe.</p> <p>No: Specifies that classes and objects follow the order defined in the derived universe. This is the default value.</p>

CORRECT_AGGREGATED_CONDITIONS_IF_DRILL

CORRECT_AGGREGATED_CONDITIONS_IF_DRILL = Yes|No

Values	Yes No
--------	----------

Default	No
Description	<p>Specifies whether Web Intelligence can aggregate measures in queries and conditions.</p> <p>Yes: Web Intelligence can aggregate measures separately in the main query and the condition, if the query is drill enabled.</p> <p>No: Web Intelligence cannot aggregate measures separately in the main query and the condition, if the query is drill enabled.</p>

CUMULATIVE_OBJECT_WHERE

CUMULATIVE_OBJECT_WHERE = Yes|No

Values	Yes No
Default	No

Description	<p>Specifies the order of WHERE clauses that have the AND connective.</p> <p>Yes: Specifies that WHERE clauses that have the AND connective are set at the end of the condition.</p> <p>No: Specifies that WHERE clauses follow standard SQL syntax.</p> <p>Example:</p> <p>If the condition is find all French clients different from John or American cities different from New York, the SQL is then:</p> <pre>WHERE (customer.first_name <> 'John') OR (city.city <> 'New York') AND customer_country.country = 'France' AND city_country.country = 'USA'</pre>
-------------	---

DECIMAL_COMMA

DECIMAL_COMMA = Yes|No

Values	Yes No
Default	No
Description	<p>Specifies that Business Objects products insert a comma as a decimal separator when necessary.</p> <p>Yes: Business Objects products insert a comma as a decimal separator when necessary.</p> <p>No: Business Objects products do not insert a comma as a decimal separator. This is the default value.</p>

DISTINCT_VALUES

DISTINCT_VALUES = GROUPBY|DISTINCT

Values	GROUPBY DISTINCT
Default	DISTINCT
Description	<p>Specifies whether SQL is generated with a DISTINCT or GROUP BY clause in a list of values and Query pane when the option "Do not retrieve duplicate rows" is active.</p> <p>DISTINCT: The SQL is generated with a DISTINCT clause, for example; SELECT DISTINCT cust_name FROM Customers</p> <p>GROUPBY: The SQL is generated with a GROUP BY clause, for example; SELECT cust_name FROM Customers GROUP BY cust_name</p>

END_SQL

END_SQL = String

Values	String
Default	<empty string>
Description	The statement specified in this parameter is added at the end of each SQL statement.
Example	<p>For IBM DB2 databases, you can use the following: END_SQL=FOR SELECT ONLY</p> <p>The server will read blocks of data much faster.</p>

EVAL_WITHOUT_PARENTHESIS

EVAL_WITHOUT_PARENTHESIS = Yes|No

Values	Yes No
Default	No
Description	<p>By default, the function @Select(Class\object) is replaced by the SELECT statement for the object <Class\object> enclosed within brackets.</p> <p>For example, when combining two @Select statements, @Select(object1) * @select(object2).</p> <p>If the SQL(object1) = A-B and SQL(object2) =C, then the operation is (A-B) * (C).</p> <p>You avoid the default adding of brackets by setting EVAL_WITHOUT_PARENTHESIS = Yes. The operation is then A - B * C.</p> <p>Yes: Brackets are removed from the SELECT statement for a function @Select(Class\object)</p> <p>No: Brackets are added around the Select statement for the function @Select(Class\object).</p>

FILTER_IN_FROM

FILTER_IN_FROM = Yes|No

Values	Yes No
Default	No

Description	<p>Determines if query conditions are included in the FROM Clause. This setting is only applicable if the other universe parameter setting ANSI92 is set to Yes.</p> <p>Yes: When editing an outer join, the default behavior property selected in the drop down list box of the Advanced Join properties dialog box in Designer, is set to "All objects in FROM".</p> <p>No: When editing an outer join, the default behavior property selected in the drop down list box of the Advanced Join properties dialog box in Designer is set to "No object in FROM".</p>
-------------	--

FIRST_LOCAL_CLASS_PRIORITY

FIRST_LOCAL_CLASS_PRIORITY = Yes|No

Values	Yes No
Default	No
Description	<p>Only taken into account when CORE_ORDER_PRIORITY=Yes.</p> <p>Yes: Classes in derived universe are placed first.</p> <p>No: Objects and sub classes from derived universe appear after those of the core universe.</p>

FORCE_SORTED_LOV

FORCE_SORTED_LOV = Yes|No

Values	Yes No
--------	--------

Default	No
Description	<p>Retrieves a list of values that is sorted.</p> <p>Yes: Specifies that the list of values is sorted.</p> <p>No: Specifies that the list of values is not sorted.</p>

INNERJOIN_IN_WHERE

INNERJOIN_IN_WHERE = Yes|No

Values	Yes No
Default	You must manually enter the parameter to activate it.
Description	<p>Allows you to force the system to generate SQL syntax with all the inner joins in the WHERE clause when ANSI92 is set to yes . This is only possible if a query contains only inner joins (Does not contain FULL OUTER, RIGHT OUTER, or LEFT OUTER joins).</p> <p>Yes: If ANSI92 is set to yes, the system generates ANSI92 join syntax in the FROM clause except when the query contains only inner joins. In this case, the inner joins go into the WHERE clause.</p> <p>No: If ANSI92 is set to Yes, the system generates ANSI 92 join syntax in the FROM clause.</p>

JOIN_BY_SQL

JOIN_BY_SQL = Yes|No

Values	Yes No
Default	No
Description	<p>Specifies how multiple SQL statements are handled. Multiple statements can be combined (provided that the database permits this).</p> <p>Yes: Specifies that multiple SQL statements are combined.</p> <p>No: Specifies that multiple SQL statements are not combined. This is the default value.</p>

MAX_INLIST_VALUES

MAX_INLIST_VALUES = 99]

Values	Integer: min 0, max depends on DB
Default	99
Description	<p>Allows you to set the maximum number of values you may enter in a condition when you use the IN LIST operator.</p> <p>99: Specifies that you may enter up to 99 values when you create a condition using the IN LIST operator. This is the default value.</p> <p>The maximum authorized value you may enter depends on your database.</p>

PATH_FINDER_OFF

Parameter is not listed by default. You must add the parameter manually to the list and set a value.

PATH_FINDER_OFF= Y|N

Values	Yes No
Default	No default. You must manually enter the parameter.
Description	Used for HPIW because the join generation is done by the database. Yes: Joins are NOT generated in the query. No: Joins are generated in the query. This is the default behavior.

REPLACE_COMMA_BY_CONCAT

REPLACE_COMMA_BY_CONCAT= Yes|No

Values	Yes No
Default	Yes
Description	In previous versions of Designer, a comma could be used to separate multiple fields in an object Select statement. The comma was treated as a concatenation operator. For universes that already use the comma in this way you can set REPLACE_COMMA_BY_CONCAT to No to keep this behavior. In the current version of Designer, this parameter is set to Yes by default, so that any expressions using a comma in this way are automatically changed to use concatenation syntax. Yes: Comma is replaced by the concatenation expression when multi field object is found. No: Keep the comma as it is.

SELFJOINS_IN_WHERE

SELFJOINS_IN_WHERE = Yes|No

Values	Yes No
Default	No
Description	<p>Self-joins are usually included in the FROM clause. This allows you to force the system to generate SQL syntax with all the conditions of a self-join in the WHERE clause. the ANSI92 parameter must be set to Yes for this parameter to be taken into account.</p> <p>You must manually add the parameter to the list to activate it.</p> <p>Yes: The conditions of a self-join go in the WHERE clause of the SQL query.</p> <p>No: The syntax for self-joins is generated according to the ANSI 92 convention, and conditions for a self-join go in the ON clause of the table join definition in the FROM clause of the SQL query.</p>

SHORTCUT_BEHAVIOR

SHORTCUT_BEHAVIOR = Global|Successive

Values	Global Successive
Default	Successive

Description	<p>Specifies how shortcut joins are applied. This parameter was formerly listed as GLOBAL_SHORTCUTS in the PRM files. The values have been changed to Global for Yes, and Successive for No.</p> <p>Global: Specifies that shortcut joins are considered one by one. A shortcut join is applied only if it really bypasses one or more tables, and if it does not remove a table from the join path used by a following shortcut join.</p> <p>Successive: Specifies that all shortcut joins are applied. Note: If it generates a Cartesian product, no shortcut joins are applied.</p>
-------------	--

THOROUGH_PARSE

THOROUGH_PARSE = Yes|No

Values	Yes No
Default	No
Description	<p>Specifies the methodology used for default Parsing in the Query pane and individual object parsing.</p> <p>Yes: PREPARE, DESCRIBE, and EXECUTE statements are used to parse SQL for objects.</p> <p>Prepare+DescribeCol+Execute</p> <p>No: PREPARE and DESCRIBE statements are used to parse SQL for objects.</p>

TRUST_CARDINALITIES

TRUST_CARDINALITIES = Yes|No

Values	Yes No
Default	No
Description	<p>Allows you to optimize the SQL in case of inflated results.</p> <p>Yes: For queries that include a measure, all conditions that inflate the measure and do not appear in the Result Objects, are transformed to sub queries to ensure that tables that may return false results for the measure are not included in the query.</p> <p>No: No optimization is implemented.</p>

UNICODE_STRINGS

UNICODE_STRINGS = Yes|No

Values	Yes No
Default	No

Description	<p>Specifies whether the current universe can manipulate Unicode strings or not. Only applies to Microsoft SQL Server and Oracle 9. If the database character set in the SBO file is set as Unicode, then it is necessary to modify the SQL generation to handle specific Unicode column types like NCHAR and NVARCHAR.</p> <p>Yes: Conditions based on strings are formatted in the SQL according to the value for a parameter UNICODE_PATTERN in the PRM file, for example for MS SQL Server (sqlsrv.prm) : UNICODE_PATTERN=N\$</p> <p>The condition Customer_name='Arai ' becomes Customer_name=N'Arai'.</p> <p>Note: When you create a prompt with @Prompt syntax based on Unicode value, the datatype should be 'U' not 'C'</p> <p>No: All conditions based on strings are formatted in the standard SQL. For example the condition Customer_name='Arai ' remains Customer_name='Arai'</p>
-------------	---



Configuring SQL generation parameters for a database



7

chapter

About SQL generation parameters

SQL generation parameters control the SQL queries that are generated to retrieve the contents of a Universe.

You can set these SQL generation parameters to operate at :

- The Universe level: you set these parameters when you create or modify a Universe.
- The database level: you set these parameters in the PRM file for the database. These parameters are overridden by any corresponding Universe-level PRM file settings.

To view the date operators, other operators, and functions available for your data access driver, open the <driver>.prm file in an XML editor.

Related Topics

- [About SQL generation parameters for a universe](#) on page 86

About PRM files

The .prm files contain parameters to control the way universes generate SQL. There is a .prm file corresponding to each database driver. The .prm files allow database-dependent factors to control how a Universe generates SQL.

You can also configure SQL generation parameters from inside a universe. The settings that you define from within a universe override .prm file settings.

The .prm file for each driver is found in the following location:

```
connectionserver-install-dir\connectionServer\<RDBMS>\ where  
<RDMS> is the database name.
```

The following <driver>.prm files are available:

Data access driver	PRM files	PRM Help text files (English)
Datafederator	datafederator.prm	datafederatoren.prm
Essbase	essbase.prm	essbaseen.prm
IBM DB2	db2udb.prm db2mvs.prm db2iseries.prm	db2udben.prm db2mvsen.prm db2iseriesen.prm
Informix	informix.prm	informixen.prm
Javabean	javabean.prm	javabeanen.prm
Microsoft SQL Server	sqlsrv7.prm for SQL Server 7 and 2000. sqlsrv.prm for SQL Server 2005. sqlsrv_as.prm for analysis services.	sqlsrv7en.prm sqlsrven.prm
Microsoft Jet	msjet.prm	msjeten.prm
MySQL	mysql.prm	mysqlen.prm
Netezza	netezza.prm	netezzaen.prm
ODBC	odbc.prm access.prm	odbcen.prm accessen.prm
Open	open.prm	openen.prm
Openaccess	openaccess.prm	openaccessen.prm
Oracle	oracle9.prm for Oracle 9. oracle.prm for Oracle 10 and higher.	oracle9en.prm oracleen.prm
Progress	progress.prm	progressen.prm

Data access driver	PRM files	PRM Help text files (English)
Red Brick	redbrick.prm	redbricken.prm
Sybase	asiq.prm sybase11.prm sybase.prm	asiqen.prm sybase11en.prm sybaseen.prm
SAP	sap.prm	sapen.prm
Teradata	teradata.prm	teradataen.prm

The <driver>.prm files are in XML format. You should use an XML editor to view and modify values in these files if necessary.

Parameter file structure

There is a <driver>.prm file for each supported RDBMS middleware. Each <driver>.prm file is divided into the following sections. Each section contains parameters that have default values set:

File section	Description
Configuration	SQL parameters used to create and optimize a universe, for example, COMMA, OUTERJOINS_GENERATION, REVERSE_TABLE_WEIGHT. These parameters are not directly available to any Business Objects product. They are described in this chapter.
DateOperations	Date operators available to Designer, Desktop Intelligence, and Web Intelligence, for example YEAR, QUARTER, MONTH.
Operators	Operators available to Designer, Desktop Intelligence, and Web Intelligence, for example ADD, SUBSTRACT, MULTIPLY.

File section	Description
Functions	<p>Functions available to Designer, Desktop Intelligence, and Web Intelligence, for example Average, Sum, Variance.</p> <p>Help text that appears when functions in this section are selected in Designer, Desktop Intelligence, and Web Intelligence is listed in the file <driver><language>.prm, for example, oracleen.prm. This file is found in the same directory as the <driver>.prm file. You can open it to view descriptions of all the functions available in the <driver>.prm file. Viewing and editing function Help text is described in the section Viewing and editing function help text on page 110.</p> <p>The Function section has the following subsections:</p> <ul style="list-style-type: none"> • Group: • ID: Name of the function. This is the name that appears in the function list in Designer. • InMacro: Specifies if the function appears in the function list in an object's Edit Select box in Designer. When value is True, the function is listed, when value is False, the function is not listed. If you set the value to True, it will be listed the next time Designer is started. • Type: Function data type. • Arguments: Arguments accepted by the function. • SQL: The SQL syntax for the function.

Note: Only the Configuration section is documented here. These parameters can be edited to optimize queries run against universes using the target data access driver.

Viewing and editing PRM file parameters

You can view, edit, and add parameters to a PRM file as follows:

1. Browse to the directory that stores the PRM file for your target data access driver. PRM files are stored here:

```
<INSTALDIR>\dataAccess\RDBMS\connectionServer\<rdbms>\<driver>.prm
```

2. Open a <driver>.prm file in a XML editor.
3. Expand sections as required.
4. Set values by entering the value in the appropriate tag.
5. Save and close the file.

Viewing and editing function help text

The Help text that appears under each function when selected in Designer is maintained in a separate XML file. You can edit and add text to describe a function by editing the file <driver><language>.prm. There is a Help text file for each language version of Business Objects products installed.

The Help text appears when a function that is stored in the PRM file is selected in Designer:

When you add a function to the PRM file, you need to add the Help text for the new function to the appropriate <driver>.<language>.prm file, for example, if you add a function to the oracle.prm file, then you also add the function name and the Help text for the function to the oracleen.prm file, if you are working with the English version of Designer.

Editing Help text for a PRM file function

You can view, edit, and add Help text for a function listed in a PRM file as follows:

1. Browse to the directory that stores the PRM language file for your target data access driver. PRM language files are stored here:

```
connectionserver-install-  
dir\connectionServer\<rdbms>\<driver><language>.prm
```

For example, oracleen.prm.

2. Open a <driver><language>.prm file in a XML editor.
3. Expand the Messages section.
4. To add Help for a new function do the following:

- Add a new section for a function. The easiest way to do this is to copy an existing function entry and copy it into the Function section. You then edit the new function text.
 - Enter Help text for the function.
5. To view or edit existing function Help text, do the following:
- Expand the Function section.
 - Expand the Message section for a function.
 - Edit Help text as required.
6. Save and close the file.

PRM file Configuration reference

The Configuration SQL parameters are listed alphabetically. To view functions, date operators, and other operators available, open a <driver>.prm file in a XML editor, each parameter is defined in the following tag:

```
<Parameter Name="parameter ">value</Parameter>
```

where parameter is the name of the parameter, and value is the value attributed to the parameter.

Each parameter is shown with the following information:

- Example of how the parameter appears in the XML file
- Description of the parameter
- Possible values that can be set for the parameter
- Default value for the parameter

Note: Certain configuration parameters must not be edited. These parameters have values set for use internally within a Business Objects product. These parameters are described in this section but contain a warning not to edit the value. You must not edit these parameters. Before editing any other PRM file parameter, you should make a backup copy of the PRM file.

BACK_QUOTE_SUPPORTED

<Parameter Name="BACK_QUOTE_SUPPORTED">Y</Parameter>

Description	Specifies whether or not to enclose table or column names containing spaces or special characters with backquotes.
Values	Y : Encloses table and column name with backquotes. N : Does not insert quotes around table and column names.
Default	Y
Result	Table name=`My Table`

CASE_SENSITIVE

<Parameter Name="CASE_SENSITIVE">N</Parameter>

Description	Specifies if the database is case sensitive. Parameter used with Oracle.
Values	Y : Specifies that the database is case sensitive. N : Specifies that the database is not case sensitive.
Default	N

CHECK_OWNER_STATE

<Parameter Name="CHECK_OWNER_STATE">N</Parameter>

Description	Specifies if the SQL checks if the database supports table classification by owner name.
Values	Y : The SQL checks if the database supports table classification by owner name. N : The SQL does not check if the database supports table classification by owner name.
Default	Y

CHECK_QUALIFIER_STATE

<Parameter Name="CHECK_QUALIFIER_STATE">N</Parameter>

Description	Specifies if the SQL checks if the database supports table classification by qualifier.
Values	Y : Checks if the database supports table classification by qualifier. N : Does not check if the database supports table classification by qualifier.
Default	Y

COMMA

<Parameter Name="COMMA">||' '||</Parameter>

Description	Specifies what database concatenation operator should be used to replaces a comma for objects that have the following syntax: Tab.Col1, Tab.Col2. Parameter is used with all data access drivers.
-------------	---

Values	' ' +' '+'
Default	' '
Result	Tab.Col1 ' ' Tab.Col2

CONCAT

<Parameter Name="CONCAT">||</Parameter>

Description	Specifies the concatenation operator. The parameter is used with all data access drivers.
Values	 +
Default	

CONSTANT_SAMPLING_SUPPORTED

CONSTANT_SAMPLING_SUPPORTED

Description	Specifies if the database supports random sampling.
Values	Yes: The database supports random sampling. No: The database does not support random sampling.
Default	Yes

DATABASE_DATE_FORMAT

```
<Parameter Name="DATABASE_DATE_FORMAT">DD-MM-YYYY  
HH24:MI:SS</Parameter>
```

Description	For Oracle only. Specifies the default date and hour formats stored on the server.
Values	DD-MM-YYYY HH24:MI:SS
Default	DD-MM-YYYY HH24:MI:SS

DATATYPE_BLOB

```
Name="Datatype_Blob">LONGVARCHAR</Parameter>
```

Description	<p>Note: Do not edit this parameter.</p> <p>Column database datatype for Blob object. This parameter is not used for this release. It is included for future compatibility.</p>
-------------	--

DATATYPE_DOUBLE

```
<Parameter Name="Datatype_Double">: Value depends on the database.
```

Description	<p>Note: Do not edit this parameter.</p> <p>Column database datatype for Date object (CREATE table command)</p>
-------------	--

DATATYPE_DTM

```
<Parameter Name="Datatype_Dtm">DATE</Parameter>
```

Description	Note: Do not edit this parameter. Column database datatype for Date object (CREATE table command)
-------------	---

DATATYPE_INT

```
<Parameter Name="Datatype_Int">NUMBER</Parameter>
```

Description	Note: Do not edit this parameter. Column database datatype for Numeric object (CREATE table command)
-------------	--

DATATYPE_NULL

```
<Parameter Name="Datatype_Null"/>
```

Description	Note: Do not edit this parameter. Database SQL syntax for Null values (INSERT command)
-------------	--

DATATYPE_STRING

```
<Parameter Name="Datatype_String">VARCHAR2</Parameter>
```

Description	<p>Note: Do not edit this parameter.</p> <p>Column database datatype for Character and Long object (CREATE table command).</p>
-------------	---

DATE_WITHOUT_QUOTE

<Parameter Name="DATE_WITHOUT_QUOTE">Y</Parameter>

Description	<p>Specifies support for dates without single-quotes in the SQL syntax.</p> <p>Parameter is used with MS Access.</p>
Values	<p>Y: Dates are not surrounded by single-quotes.</p> <p>N: Dates are surrounded by single-quotes.</p>
Default	Y

EXT_JOIN

<Parameter Name="EXT_JOIN">YES</Parameter>

Description	<p>Specifies if outer joins are supported.</p> <p>This parameter is used by all data access drivers.</p>
Values	<p>YES: The database supports outer joins.</p> <p>NO: The database does not support outer joins. The Outer join check boxes in the Designer's Edit Join dialog box are dimmed.</p>
Default	YES

EXT_JOIN_INVERT

<Parameter Name="EXT_JOIN_INVERT">YES</Parameter>

Description	Specifies how to display an outer join symbol in a join expression. This parameter is used with IBM DB2, Informix, Oracle, and Teradata.
Values	YES: When you click an Outer join check box in the Edit Join dialog box of Designer, the outer join symbol appears reversed in position in a join expression. NO: When you click an Outer join check box in the Edit Join dialog box of Designer, the outer join symbol appears on the same side on which you created the outer join.
Default	YES

EXTERN_SORT_EXCLUDE_DISTINCT

<Parameter Name="EXTERN_SORT_EXCLUDE_DISTINCT">Y</Parameter>

Description	Specifies whether or not the application generates a SELECT DISTINCT when a query contains an ORDER BY .
Values	Y: A SELECT DISTINCT is not generated when the query contains an ORDER BY . N: A DISTINCT is generated when the query contains an ORDER BY .
Default	Y

GROUPBY_EXCLUDE_COMPLEX

Parameter Name="GROUPBY_EXCLUDE_COMPLEX">N</Parameter>

Description	Specifies whether the database allows you to enter formulas, aliases or indexes in GROUP BY clauses. Parameter is used with IBM DB2.
Values	Y: Specifies that the database does not allow you to enter formulas, aliases or indexes in GROUP BY clauses. If you run a query containing measure objects and complex objects (e.g. with the substring function or the concatenation operator), your Business Objects product displays the following error message: Your database does not allow you to do aggregations with the <object name> object. N: Specifies that the database does allow you to enter formulas, aliases or indexes in GROUP BY clauses.
Default	N

GROUPBY_WITH_ALIAS

<Parameter Name="GROUPBY_WITH_ALIAS">Y</Parameter>

Description	Specifies whether the database can create a GROUP BY clause in the SELECT statement. Parameter is used with Red Brick.
Values	Y: Allows you to create a GROUP BY clause in the SELECT statement. An alias would be used for example to replace a T1.col + T2.col +...Tn.col n .col statement. N: Does not let you to create a GROUP BY clause in the SELECT statement.
Default	Y

GROUPBY_WITHOUT_CONSTANT

<Parameter Name="GROUPBY_WITHOUT_CONSTANT">Y</Parameter>

Description	Specifies whether or not you authorize the addition of objects whose SQL definition is a constant in a GROUP BY clause. Parameter is used by IBM DB2 and Microsoft SQL Server
Values	Y: Specifies that you can add any constant object to the query but it will not be present in the GROUP BY clause. N: Specifies that you can insert all the objects of a query (i.e. without aggregate functions) in a GROUP BY clause.
Default	Y

GROUPBYCOL

<Parameter Name="GROUPBYCOL">NO</Parameter>

Description	Specifies whether a GROUP BY clause accepts integers for column names.
Values	YES: Specifies that a GROUP BY clause accepts a column's index from the SELECT instead of column's name. NO: Specifies that a GROUP BY clause does not accepts a column's index from the SELECT instead of column's name.
Default	NO

IDENTIFIER_DELIMITER

<Parameter Name="IDENTIFIER_DELIMITER">"</Parameter>

Description	<p>Specifies that table or column names that contain spaces or special characters are enclosed within quotation marks if the parameter BACK_QUOTE_SUPPORTED is activated.</p> <p>To use this parameter, BACK_QUOTE_SUPPORTED must be set to BACK_QUOTE_SUPPORTED=Y. This is the default value.</p>
Values	<p>": Table or column names that contain spaces or special characters are enclosed in double quotation marks.</p> <p>': Table or column names that contain spaces or special characters are enclosed in single quotation marks. This value can be used only with Microsoft Access.</p>
Default	"
Result	Table name="My Table"

IF_NULL

<Parameter Name="IF_NULL">N0</Parameter>

Description	Specifies a function that takes two parameters. If the first parameter returns NULL, the second parameter value is used.
Values	Database-dependent.
Default	Database-dependent.

INTERSECT

<Parameter Name="INTERSECT">INTERSECT</Parameter>

Description	Specifies if the database supports the INTERSECT set operator.
Values	INTERSECT: The database supports the INTERSECT set operator. No value: The database does not support the INTERSECT set operator. In this case, two queries are generated.
Default	INTERSECT

KEY_INFO_SUPPORTED

<Parameter Name="KEY_INFO_SUPPORTED">Y</Parameter>

Description	Specifies if you can retrieve primary and secondary key definitions from the database account.
Values	Y: Specifies that the database lets you retrieve primary and secondary key definitions from the database account. This parameter enables Designer to display the keys in the Structure window. N: Specifies that the database does not let you retrieve primary and secondary key definitions from the database account.
Default	Y

LEFT_OUTER

<Parameter Name="LEFT_OUTER">\$(<+></Parameter>

<Parameter Name="LEFT_OUTER">\$*</Parameter>

Description	Specifies the left outer join syntax.
-------------	---------------------------------------

Values	\$(+) This syntax is used with Oracle. \$ represents a join expression. \$* This syntax is used with Sybase, MS SQL Server and Red Brick. \$ represents a join expression.
Default	See values above.

LENMAXFORCOLUMNNAME

<Parameter Name="LenMaxForColumnName">30</Parameter>

Description	Note: Do not edit this parameter. Maximum length for column name (by default the object name is proposed) (CREATE table command)
-------------	--

LENMAXFORTABLENAME

<Parameter Name="LenMaxForTableName">30</Parameter>

Description	Note: Do not edit this parameter. Maximum length for VARCHAR column datatype (CREATE table command).
-------------	---

LENMAXFORVARCHAR

<Parameter Name="LenMaxForVarchar">254</Parameter>

Description	<p>Note: Do not edit this parameter.</p> <p>Maximum length for VARCHAR column datatype (CREATE table command).</p>
-------------	---

MINUS

<Parameter Name="MINUS">MINUS</Parameter>

Description	Specifies if the database supports the MINUS set operator.
Values	<p>MINUS Specifies that the database supports the MINUS set operator.</p> <p>EXCEPT Specifies that the database supports the MINUS set operator.</p> <p>no value Specifies that the database does not support the MINUS set operator. In this case, two queries are generated.</p>
Default	MINUS

NO_DISTINCT

<Parameter Name="NO_DISTINCT">Y</Parameter>

Description	<p>Specifies if the database supports the DISTINCT keyword.</p> <p>This parameter is used with MS Access.</p>
-------------	--

Values	<p>Y: Specifies that the database does not support the DISTINCT keyword. This behavior disables:</p> <ul style="list-style-type: none"> The Distinct Values option that appears when you click the View Values button in the Quick Design wizard. The Countdistinct function that appears when you create a condition with the Calculation operand in the Query Panel. <p>N: Specifies that the database does support the DISTINCT keyword.</p>
Default	Y

NULL_IN_SELECT_SUPPORTED

<Parameter Name="NULL_IN_SELECT_SUPPORTED">Yes</Parameter>

Description	Specifies if the database supports NULL as a column in the SELECT statement.
Values	<p>Yes: NULL is supported as a column in the SELECT statement.</p> <p>No: NULL is not supported as a column in the SELECT statement.</p>
Default	Yes

OLAP_CLAUSE

<Parameter Name="OLAP_CLAUSE">WHEN</Parameter>

Description	Specifies whether Business Objects products generate a WHEN or QUALIFY clause if a function listed in the RISQL_Functions parameter is used in a condition. This parameter must be used with the GROUPBY clause.
Values	WHEN: Generates a WHEN clause if a function listed in the RISQL_Functions parameter is used in a condition. This is the default value for Red Brick databases. QUALIFY: Generates a QUALIFY clause if a function listed in the RISQL_Functions parameter is used in a condition. This is the default value for Teradata databases.
Default	See values above.

OUTERJOINS_GENERATION

```
<Parameter Name="OUTERJOINS_GENERATION">ANSI92</Parameter>
```

This parameter controls the default outer join generation behavior. You can set that

- Outer join generation conforms to the **ANSI92** specification.
- Outer join generation remains the same as for previous versions of Designer.

Note: The PRM file OUTERJOINS_GENERATION parameter relates to the universe ANSI92 setting in the following way:

- If the PRM file OUTERJOINS_GENERATION parameter is set to **ANSI92** and the universe ANSI92 setting is set to **NO**, the PRM parameter overrides the universe setting and outer joins conform to ANSI92 behavior.
- If the PRM file OUTERJOINS_GENERATION parameter is set to **USUAL**, then the universe ANSI92 setting takes precedence, and outer joins conform to ANSI92 depending on whether the universe ANSI92 setting is **YES** or **NO**.

Description	<p>Specifies the SQL syntax for outer joins.</p> <p>The value ANSI_92 generates an outer join in the FROM clause. Other values generate the outer join in the WHERE clause.</p> <p>When you modify this setting, you should check join properties to verify that the outer join expression is valid, and that the cardinalities are correct. ANSI92 does not support any manual customization in the join syntax.</p>
Values	<p>The primary values for OUTERJOINS_GENERATION are:</p> <ul style="list-style-type: none"> • ANSI92: The default outer join behavior conforms to the ANSI92 standard no matter what the value of the Designer-level ANSI92 setting. • No: Outer joins are not supported. • USUAL: The default outer join behavior is the same as with previous versions of Designer. This behavior is overridden if the Designer-level ANSI92 parameter is set to Yes. <p>Other settings are available depending on the database. See the defaults below.</p>
Default	<p>ANSI_92: Default value for Oracle, SQL Server 2005 and Sybase.</p> <p>DB2: Default value for IBM DB2.</p> <p>FULL_ODBC: Can be used with Microsoft SQL Server</p> <p>INFORMIX Default value for Informix.</p> <p>INGRES Default value for Teradata.</p> <p>NO: Default value for ODBC.</p> <p>USUAL: Default value for Neoview, Netezza, Red Brick, SQL Server 2000.</p>

Examples of **OUTERJOINS_GENERATION** parameter settings

Setting = **USUAL**:

```
FROM T1, T2
WHERE T1.col1(+) = T2.col2
```

Setting = DB2:

```
FROM T2 LEFT OUTER JOIN T1
      ON T1.col1 = T2.col2
```

Setting = ODBC:

```
FROM {oj T1 LEFT OUTER JOIN T2 ON T1.col1=T2.col2}
      Where (T2.col3 = T3.col1)
```

Setting = INFORMIX

```
FROM T2
OUTER T1
WHERE T1.col1=T2.col2
```

Setting = FULL-ODBC

```
FROM {oj T1 RIGHT OUTER JOIN T2 ON T2.col2=T1.col1
      T2 INNER JOIN 3 on T2.col3 = T3.col1}
```

Setting = ANSI_92:

```
SELECT DISTINCT
      t1.col1,
      t2.col2
FROM
      (t1 RIGHT OUTER JOIN t2 ON (t1.col1=t2.col2) )
```

Using OUTERJOINS with Oracle

The default `OUTERJOINS_GENERATION` setting (**ANSI92**) can affect the behavior of existing universes irrespective of the universe-level setting for the **ANSI92** parameter.

To set that your existing Oracle universes behave as with the previous Designer versions:

1. In the PRM file, ensure that the `OUTERJOINS_GENERATION` parameter is set to **USUAL**.
2. In the PRM file , set the **LEFT_OUTER** and **RIGHT_OUTER** parameters to `$(+)`

Related Topics

- [LEFT_OUTER](#) on page 122
- [RIGHT_OUTER](#) on page 133
- [About SQL generation parameters for a universe](#) on page 86

- [Universe SQL parameters reference](#) on page 87

OVER_CLAUSE

<Parameter Name="OVER_CLAUSE">Y</Parameter>

Description	Allows Business Objects products to include RISQL functions when generating SQL. The supported RISQL functions for the database are listed in the RISQL_FUNCTIONS parameter.
Values	Y: BusinessObjects products can include RISQL functions when generating SQL. N: BusinessObjects products cannot include RISQL functions when generating SQL.
Default	Y

OWNER

<Parameter Name="OWNER">Y</Parameter>

Description	Specifies if the database authorizes to prefix tables by the owner name.
Values	Y: Specifies that the database supports prefixing tables by the owner name. N: Specifies that the database does not support prefixing tables by the owner name.
Default	Y

PERCENT_RANK_SUPPORTED

<Parameter Name="PERCENT_RANK_SUPPORTED">Yes</Parameter>

Description	Specifies if the Percent Rank analytical function is supported by the database.
Values	Yes: the Percent Rank analytical function is supported by the database. No: the Percent Rank analytical function is not supported by the database.
Default	Yes

PREFIX_SYS_TABLE

<Parameter Name="PREFIX_SYS_TABLE">RBW_</Parameter>

<Parameter Name="PREFIX_SYS_TABLE">MSys</Parameter>

Description	Specifies if the system tables are displayed in Designer. This parameter is found in the <code>access.prm</code> file and <code>redbrick.prm</code> file.
Values	MSys: Specifies that the MS Access system tables are hidden in the Designer table browser. Default value for MS Access. RBW_: Specifies that the Red Brick system tables are hidden in the Designer table browser. Default value for Red Brick. no value: Specifies that the database system tables are displayed in the Designer table browser.
Default	See values above.

QUALIFIER

<Parameter Name="QUALIFIER">N</Parameter>

Description	Specifies whether the database authorizes to prefix tables by the qualifier name.
Values	Y: Specifies that the database does support prefixing tables by the qualifier name. N: Specifies that the database does not support prefixing tables by the qualifier name.
Default	RDBMS dependant.

QUOTE_OWNER

<Parameter Name="QUOTE_OWNER">Y</Parameter>

Description	Specifies whether or not an owner name should be in single quotes. Used by Informix only.
Values	Y: Specifies that table names are prefixed by an owner name in single quotes. This is mandatory for an ANSI compliant Informix database. If not, Informix converts the owner name to upper case characters. N: Specifies that table names are not prefixed by an owner name in single quotes.
Default	Y
Result	SELECT Alias.col (<Alias> is a local Alias) FROM 'Owner'.table.col Alias

RANK_SUPPORTED

```
<Parameter Name="RANK_SUPPORTED">Yes</Parameter>
```

Description	Specifies if the Rank analytical function is supported by the database.
Values	Yes: the Rank analytical function is supported by the database. No: the Rank analytical function is not supported by the database.
Default	Yes

REFRESH_COLUMNS_TYPE

```
<Parameter Name="REFRESH_COLUMNS_TYPE">0</Parameter>
```

Description	Indicates how columns are refreshed.
Values	O: Columns are refreshed by owner name. This is the default value with Oracle. Q: Columns are refreshed by qualifier name. This is the default value with Red Brick, Sybase, MS SQL Server and MS Access. T: Columns are refreshed by table name.
Default	See Values above.

REVERSE_TABLE_WEIGHT

```
<Parameter Name="REVERSE_TABLE_WEIGHT">Y</Parameter>
```

Description	Specifies in which order tables are to be generated. This parameter is used with Oracle. This parameter can also be used with some other databases, possibly with the Y and N reversed. This parameter is not supported for Teradata.
Values	N: Specifies that tables are generated from the largest to the smallest. Y: Specifies that tables are generated from the smallest to the largest.
Default	Y

RIGHT_OUTER

```
<Parameter Name="RIGHT_OUTER">$(<+>)</Parameter>
```

```
<Parameter Name="RIGHT_OUTER">*$</Parameter>
```

Description	Specifies the right outer join syntax.
Values	\$(<+>): Used with Oracle. \$ represents a join expression. *\$: Used with Sybase, MS SQL Server and Red Brick. \$ represents a join expression.
Default	See values above.

RISQL_FUNCTIONS

```
<Parameter Name="RISQL_FUNCTIONS">RANK,SUM,AVG,COUNT,MIN,MAX</Parameter>
```

Description	Lists the RISQL functions supported by the database.
-------------	--

Values	<ul style="list-style-type: none"> • Oracle default list: RANK, SUM, AVG, COUNT, MIN, MAX, RATIO_TO_REPORT, CUME_DIST, CORR, DENSE_RANK, FIRST_VALUE, LAST_VALUE, LAG, LEAD, NTILE, PERCENT_RANK, PERCENTILE_CONT, PERCENTILE_DISC, ROW_NUMBER, CORR, COVAR_POP, COVAR_SAMP, REGR_SLOPE, REGR_INTERCEPT, REGR_COUNT, REGR_R2, REGR_AVGX, REGR_AVGY, REGR_SXX, REGR_SYY, REGR_SXY, STDDEV, STDDEV_POP, STDDEV_SAMP, VAR_POP, VAR_SAMP, VARIANCE • BM DB2 default list: RANK, SUM, AVG, COUNT, MIN, MAX • Teradata default list: CSUM, MAVG, MDIFF, MLIN REG, MSUM, RANK, QUANTILE • Red Brick default list: CUME, MOVINGAVG, MOVING SUM, RANK, RATIO_TO_REPORT, TERTILE
Default	See values above.

SEED_SAMPLING_SUPPORTED

<Parameter Name="SEED_SAMPLING_SUPPORTED">Yes</Parameter>

Description	Specifies if seed-based random sampling capability is supported by the database.
Values	Yes: seed-based sampling is supported by the database. No: seed-based sampling is not supported by the database.
Default	No

SORT_BY_NO

<Parameter Name="SORT_BY_NO">NO</Parameter>

Description	Specifies if users are authorized to sort on columns (represented as objects in the universe) that are not included in the SELECT statement.
Values	<p>YES: Specifies that users are not authorized to sort on columns if they are not included in the SELECT statement. When the parameter is set to YES, the Manage Sorts button is dimmed in the Query Panel.</p> <p>NO: Specifies that you are authorized to sort on columns even if they are not included in the SELECT statement.</p>
Default	NO

UNICODE_PATTERN

<Parameter Name="UNICODE_PATTERN">UNISTR(\$)</Parameter>

Description	<p>SQL Server and Oracle only.</p> <p>Only applies when the universe SQL generation parameter UNICODE_STRINGS is set to Y. All conditions based on strings are then formatted with this string value.</p>
Default	<p>Microsoft SQL Server: N\$</p> <p>Oracle: UNISTR(\$)</p>
Example	UNICODE_PATTERN=N\$

UNION

<Parameter Name="UNION">UNION</Parameter>

Description	Specifies if the database supports the UNION set operator.
Values	UNION: The database supports the UNION set operator. no value : The database does not support the UNION set operator. In this case, two queries are generated.
Default	UNION

USER_INPUT_DATE_FORMAT


```
<Parameter Name="USER_INPUT_DATE_FORMAT">'dd-MM-yyyy  
HH:mm:ss'</Parameter>
```

Description	Specifies the default date and hour formats generated in the WHERE clause of a SQL script.
Values	<p>{d 'yyyy-mm-dd'} This is the default date format with ODBC.</p> <p>'DD-MM-YYYY HH:MM:SS' This is the default date and hour formats with Oracle.</p> <p>'YYYY-MM-DD HH:MM:SS' This is the default date and hour formats with Informix.</p> <p>'yyyy-mm-dd HH:mm:ss' This is the default date and hour formats with MS SQL Server and for most IBM DB2 servers.</p> <p>'mm/dd/yyyy hh:m:s am/pm' This is the default date and hour formats with Sybase.</p> <p>'yyyy-mm-dd' This is the default date format with a Sybase gateway.</p> <p>Note: If you need to use time or timestamp variables with ODBC, you must replace the default date format value with: {t 'hh:mm:ss'} or {t/s 'yyyy-mm-dd hh:mm:ss'} in the odbc.sbo file.</p>
Default	See values above.

USER_INPUT_NUMERIC_SEPARATOR

<Parameter Name="USER_INPUT_NUMERIC_SEPARATOR">.</Parameter>

Description	Specifies the default decimal separator that is used in the generated SQL script.
Values	'.'
Default	'.'



Data type conversion reference



8

chapter



Data type conversion

This chapter contains data conversion tables for each supported RDBMS. Business Objects supports four datatypes:

- date
- character
- number
- long text

Data conversion tables are available for the following RDBMS:

- IBM DB2
- IBM DB2
- Informix
- Microsoft SQL Server
- Oracle
- Red Brick
- Sybase
- Teradata

Note: Data conversion tables for other supported RDBMS will be available in future updates of the Data Access guide. The Data Access guide and other Business Objects documentation is available from the product documentation page of the Business Objects Customer Support site: <http://support.businessobjects.com/documentation>

IBM DB2

The following table lists the IBM DB2 internal data types and their equivalent in Business Objects products:

IBM/DB2 datatype	BusinessObjects datatype
CLOB	LONG TEXT

IBM/DB2 datatype	BusinessObjects datatype
CHARACTER	CHARACTER
DATE	DATE
DECIMAL	NUMBER
DOUBLE	NUMBER
FLOAT	NUMBER
INTEGER	NUMBER
LONG VARCHAR	LONG TEXT
NUMERIC	NUMBER
SMALLINT	NUMBER
TIME	DATE
TIMESTAMP	DATE
VARCHAR	CHARACTER

Informix

The following table lists the Informix internal data types and the equivalent in Business Objects products:

Informix datatype	BusinessObjects datatype
CHAR	CHARACTER
DATE	DATE
DATETIME	DATE

Informix datatype	BusinessObjects datatype
DECIMAL	NUMBER
FLOAT	NUMBER
INTEGER	NUMBER
MONEY	NUMBER
NCHAR	CHARACTER
NVARCHAR	CHARACTER
SERIAL	NUMBER
SMALLINT	NUMBER
SMALLFLOAT	NUMBER
TEXT	LONG TEXT
VARCHAR	CHARACTER

Microsoft SQL Server

The following table lists the Microsoft SQL Server internal data types and their BusinessObjects equivalent.

Microsoft SQL Server datatype	BusinessObjects datatype
BIT	NUMBER
BOOLEAN	NUMBER
CHAR	CHARACTER
DATETIME	DATE

Microsoft SQL Server datatype	BusinessObjects datatype
DECIMAL	NUMBER
FLOAT	NUMBER
INT	NUMBER
MONEY	NUMBER
NUMERIC	NUMBER
REAL	NUMBER
SMALLDATETIME	DATE
SMALLINT	NUMBER
SMALLMONEY	NUMBER
TEXT	CHARACTER
TINYINT	NUMBER
VARCHAR	CHARACTER

Oracle

The following table lists the Oracle internal datatypes and their equivalent in Business Objects products:

Oracle datatype	BusinessObjects datatype
CHAR	CHARACTER
CLOB	LONG TEXT
DATE	DATE

Oracle datatype	BusinessObjects datatype
FLOAT	NUMBER
INTEGER	NUMBER
LONG	LONG TEXT
NUMBER	NUMBER
VARCHAR	CHARACTER
VARCHAR2	CHARACTER

Red Brick

The following table lists the Red Brick internal data types and their BusinessObjects equivalent.

Red Brick datatype	BusinessObjects datatype
CHAR	CHARACTER
DATE	DATE
DECIMAL	NUMBER
FLOAT	NUMBER
INTEGER	NUMBER
NUMERIC	NUMBER
SMALLINT	NUMBER
TIME	DATE
TIMESTAMP	DATE

TIME and TIMESTAMP support in a WHERE clause

Objects that use the TIME and TIMESTAMP data type are not supported in a WHERE clause for queries run against a Red Brick database. You can set the properties of an object in a universe to support the use of TIME and TIMESTAMP in the WHERE clause as follows:

To support TIME or TIMESTAMP datatype object in WHERE clause:

1. From Designer, double-click an object in the Universe pane.

The Edit Properties of Object dialog box appears.

2. Click the Definition tab.
3. Select Date from the Type drop down list.
4. Click the Advanced tab.
5. In the Database Format text box type the following information:

For...	Type
TIME support	HH:MM:SS
TIMESTAMP support	MM/DD/YYYY HH:MM:SS

6. Click **OK**.

Sybase

The following table lists the Sybase internal datatypes and their equivalent in Business Objects products.

Sybase datatype	BusinessObjects datatype
BIT	NUMBER
CHAR	CHARACTER
DATETIME	DATE

Sybase datatype	BusinessObjects datatype
DECIMAL	NUMBER
FLOAT	NUMBER
INT	NUMBER
MONEY	NUMBER
NUMERIC	NUMBER
REAL	NUMBER
SMALLDATETIME	DATE
SMALLINT	NUMBER
SMALLMONEY	NUMBER
TINYINT	NUMBER
VARCHAR	CHARACTER

Teradata

The following table lists the Teradata internal datatypes and their equivalent in Business Objects products.

Teradata datatype	BusinessObjects datatype
DATE	DATE
DECIMAL	NUMBER
FLOAT	NUMBER
INTEGER	NUMBER

Teradata datatype	BusinessObjects datatype
SMALLINT	NUMBER
BYTEINT	NUMBER
DATETIME	DATE
VARCHAR	CHARACTER



Business Objects information resources



9

chapter



Documentation and support

Business Objects offers a full documentation set covering its products and their deployment. Additional support and services are also available to help maximize the return on your business intelligence investment. The following sections detail where to get Business Objects documentation and how to use the resources at Business Objects to meet your needs for technical support, education, and consulting.

Useful addresses at a glance

Address	Content
<i>Business Objects product information</i> http://www.businessobjects.com	Information about the full range of Business Objects products.
<i>Product documentation</i> http://www.support.businessobjects.com/documentation/product_guides/default.asp	Business Objects product documentation, including the Business Objects Documentation Roadmap.
<i>Business Objects Documentation mailbox</i> documentation@businessobjects.com	Send us feedback or questions about documentation.
<i>Knowledge Base (KB) and Knowledge Exchange</i> www.techsupport.businessobjects.com	Technical articles, documents, case resolutions, plus an online forum.
<i>Online Customer Support</i> http://www.support.businessobjects.com	Information on Customer Support programs, as well as links to technical articles, downloads, and online forums.
<i>Business Objects Consulting Services</i> http://www.businessobjects.com/services/consulting/	Information on how Business Objects can help maximize your business intelligence investment.

Address	Content
<i>Business Objects Education Services</i> http://www.businessobjects.com/services/training	Information on Business Objects training options and modules.

Documentation

You can find answers to your questions on how to install, configure, deploy, and use Business Objects products from the documentation.

What's in the documentation set?

View or download the *Business Objects Documentation Roadmap*, available with the product documentation at <http://www.businessobjects.com/support/>.

The Documentation Roadmap references all Business Objects guides and lets you see at a glance what information is available, from where, and in what format.

Where is the documentation?

You can access electronic documentation at any time from the product interface, the web, or from your product CD.

- Documentation from the products

Online help and guides in Adobe PDF format are available from the product Help menus. Where only online help is provided, the online help file contains the entire contents of the PDF version of the guide.

- Documentation on the web

The full electronic documentation set is available to customers on the web from support website at: <http://www.support.businessobjects.com/>.

- Documentation on the product CD

Look in the docs directory of your product CD for versions of guides in Adobe PDF format.

Send us your feedback

Do you have a suggestion on how we can improve our documentation? Is there something you particularly like or have found useful? Email us, and we will do our best to ensure that your suggestion is included in the next release of our documentation: documentation@businessobjects.com.

Note: If your issue concerns a Business Objects product and not the documentation, please contact our Customer Support experts. For information about Customer Support visit: <http://www.businessobjects.com/support>.

Customer support, consulting and training

A global network of Business Objects technology experts provides customer support, education, and consulting to ensure maximum business intelligence benefit to your business.

How can we support you?

Business Objects offers customer support plans to best suit the size and requirements of your deployment. We operate customer support centers in the following countries:

- USA
- Australia
- Canada
- United Kingdom
- Japan

Online Customer Support

The Business Objects Customer Support website contains information about Customer Support programs and services. It also has links to a wide range of technical information including knowledgebase articles, downloads, and support forums.

<http://www.businessobjects.com/support/>

Looking for the best deployment solution for your company?

Business Objects consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in relational and multidimensional databases, in connectivities, database design tools, customized embedding technology, and more.

For more information, contact your local sales office, or contact us at:

<http://www.businessobjects.com/services/consulting/>

Looking for training options?

From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style. Find more information on the Business Objects Education website:

<http://www.businessobjects.com/services/training>



Get More Help



appendix



Online documentation library

Business Objects offers a full documentation set covering all products and their deployment. The online documentation library has the most up-to-date version of the Business Objects product documentation. You can browse the library contents, do full-text searches, read guides on line, and download PDF versions. The library is updated regularly with new content as it becomes available.

http://support.businessobjects.com/documentation/product_guides/

Additional developer resources

<http://devlibrary.businessobjects.com>

Online customer support

The Business Objects Customer Support web site contains information about Customer Support programs and services. It also has links to a wide range of technical information including knowledgebase articles, downloads, and support forums.

<http://www.businessobjects.com/support/>

Looking for the best deployment solution for your company?

Business Objects consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in relational and multidimensional databases, in connectivities, database design tools, customized embedding technology, and more.

For more information, contact your local sales office, or contact us at:

<http://www.businessobjects.com/services/consulting/>

Looking for training options?

From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style. Find more information on the Business Objects Education web site:

<http://www.businessobjects.com/services/training>

Send us your feedback

Do you have a suggestion on how we can improve our documentation? Is there something you particularly like or have found useful? Drop us a line, and we will do our best to ensure that your suggestion is included in the next release of our documentation:

<mailto:documentation@businessobjects.com>

Note: If your issue concerns a Business Objects product and not the documentation, please contact our Customer Support experts. For information about Customer Support visit: <http://www.businessobjects.com/support/>.

Business Objects product information

For information about the full range of Business Objects products, visit: <http://www.businessobjects.com>.

Index

A

- access
 - New Connection wizard 28
- ANSI92
 - universe parameter 87
- ANSI92 SQL syntax 126
- architecture, system 11
- Array Bind Available 58
- Array Bind Size 58, 59, 60, 68, 70, 72
- Array Fetch Available 58
- Array Fetch Size 59
- AUTO_UPDATE_QUERY
 - universe parameter 87

B

- BACK_QUOTE_SUPPORTED 112
- Blob_Comparison 112
- BLOB_COMPARISON
 - universe parameter 87
- Boundary_Weight_Table 112
- BOUNDARY_WEIGHT_TABLE
 - universe parameter 87
- Business Objects
 - consulting services 47, 153
 - support services 153
 - training services 47, 153

C

- Case_Sensitive 112
- CASE_SENSITIVE 112
- CFG file
 - Charset List Extension 44

- CFG file (*continued*)
 - Config File Extension 45
 - Description Extension 45
 - Distribution section 50
 - Driver defaults section 43
 - editing 43
 - Enable Failed Load 45
 - Load Drivers On Startup 46
 - Lookup parameters 50
 - Max Pool Time 47
 - Protocols parameters 50
 - settings parameters reference 44
 - SQL External Extension 48
 - SQL Parameter Extension 48
 - Strategies Extension 48
 - viewing 43
- CharSet 60
- Charset List Extension
 - CFG file 44
- CHECK_OWNER_STATE 112
- CHECK_QUALIFIER_STATE 113
- checking database access 25
- ClassPath, creating a JavaBean connection 38
- COLUMNS_SORT
 - universe parameter 87
- COMBINE_WITHOUT_PARENTHESES
 - universe parameter 87
- COMBINED_WITH_SYNCHRO
 - universe parameter 87
- COMMA 113
- COMPARE_CONTEXTS_WITH_JOINS
 - universe parameter 87
- CONCAT 114
- Config File Extension
 - CFG file 45

Index

- configuration files
 - installed SBO files 53
 - JavaBean SBO example 39
 - parameter categories 56
 - connection
 - creation process overview 16
 - parameters overview 28
 - personal 30
 - secured 30
 - shared 30
 - start creating 28
 - connection components 10
 - Connection Server
 - definition 10
 - in standalone server mode 50
 - consultants, Business Objects 153
 - CORE_ORDER_PRIORITY
 - universe parameter 87
 - CORRECT_AGGREGATED_CONDITIONS_IF_DRILL
 - universe parameter 87
 - Cost Estimate Unit 117
 - create
 - connection overview 16
 - dynamic SQL parameters 86
 - cs.cfg
 - about 42
 - cscheck function list 18
 - cscheck syntax 18
 - cscheck, getting help 17
 - Cumulative_Object_Where 62, 117
 - CUMULATIVE_OBJECT_WHERE
 - universe parameter 87
 - CursorForward 117
 - customer support 153
- ## D
- data access
 - driver level files 13
 - files installed 12
 - global files 13
 - installed files 12
 - data access driver
 - checking availability on a machine 23
 - driver
 - checking availability on a machine 23
 - driver, checking installed data access drivers 23
 - files used by 13
 - installation
 - checking data access driver install 23
 - returning a list of available drivers 20
 - data access drivers, about 11
 - database access, checking 25
 - database engines, listing supported 19
 - DATABASE_DATE_FORMAT 115
 - datatype
 - IBM DB2 140
 - Informix 141
 - Oracle 143
 - Red Brick 144
 - SQL Server 142
 - Sybase 145
 - Teradata 146
 - DATATYPE_BLOB 115
 - DATATYPE_DOUBLE 115
 - DATATYPE_DTM 116
 - DATATYPE_INT 116
 - DATATYPE_NULL 116
 - DATATYPE_STRING 116
 - Date_Without_Quote 117
 - DATE_WITHOUT_QUOTE 117
 - DECIMAL_COMMA
 - universe parameter 87
 - delete
 - SQL parameters 86
 - Description Extension
 - CFG file 45
 - Description File 62
 - DISTINCT_VALUES
 - universe parameter 87
 - Distribution
 - CFG file 50

- documentation
 - feedback on [152](#)
 - on product CD [151](#)
 - on the web [151](#)
 - roadmap [151](#)
- Driver defaults
 - CFG file [43](#)
- driver.sbo parameters
 - Cost Estimate Unit [117](#)
 - CursorForward [117](#)
 - Date_Without_Quote [117](#)
 - Lock Mode [80](#)
 - LongVarcharNotSupported [80](#)
- driveren.prm parameters
 - Blob_Comparison [112](#)
 - Boundary_Weight_Table [112](#)
 - Case_Sensitive [112](#)
 - Cumulative_Object_Where [62](#), [117](#)
 - Ext_Join [117](#)
 - Intersect [121](#)
 - Left_Outer [122](#)
 - Max_Inlist_Values [124](#)
 - Minus [124](#)
 - OLAP_Clause [125](#)
 - Quote_Owner [131](#)
 - Refresh_Columns_Type [132](#)
- dynamic
 - SQL parameters [86](#)

E

- edit
 - dynamic SQL parameters [86](#)
 - PRM file parameters [109](#)
- education. See training [47](#)
- Enable Failed Load
 - CFG file [45](#)
- END_SQL
 - universe parameter [87](#)
- EVAL_WITHOUT_PARENTHESES
 - universe parameter [87](#)
- Ext_Join [117](#)

- EXT_JOIN [117](#)
- EXT_JOIN_INVERT [118](#)
- EXTERN_SORT_EXCLUDE_DISTINCT [118](#)

F

- Family [63](#)
- feedback, on documentation [152](#)
- FILTER_IN_FROM
 - universe parameter [87](#)
- FIRST_LOCAL_CLASS_PRIORITY
 - universe parameter [87](#)
- FORCE_SORTED_LOV
 - universe parameter [87](#), [114](#)
- ForeignKeys Available [78](#)
- functions, cscheck [18](#)

G

- generate
 - dynamic SQL parameters [86](#)
- global
 - data access files [13](#)
- GROUPBY_EXCLUDE_COMPLEX [119](#)
- GROUPBY_WITH_ALIAS [119](#)
- GROUPBY_WITHOUT_CONSTANT [120](#)
- GROUPBYCOL [120](#), [121](#), [125](#), [130](#), [132](#), [134](#)

H

- help, cscheck [17](#)

I

- IBM DB2
 - datatypes [140](#)
- information resources [150](#)
- Informix
 - datatypes [141](#)
- Informix configuration parameters [75](#)

Index

infrastructure, running the cscheck tool 18
INNERJOIN_IN_WHERE
 universe parameter 87
installation
 files installed 12
installation directory
 location 42
Intersect 121
INTERSECT 121

J

JavaBean connectins, about 38
JavaBean connection
 creating 38
JavaBean SBO example 39
JavaBean SBO parameters 77
JDBC connection
 configuring
 Java drivers
 configuring JDBC connections 36
 creating 36
jdbc.sbo 36
JOIN_BY_SQL
 universe parameter 87

K

KEY_INFO_SUPPORTED 122
Knowledge Base 47

L

Left_Outer 122
LEFT_OUTER 122
LENMAXFORCOLUMNNAME 123
LENMAXFORTABLENAME 123
LENMAXFORVARCHAR 123
Load Drivers On Startup
 CFG file 46
Lock Mode 80

LongVarcharNotSupported 80
Lookup
 CFG file 50

M

Max Pool Time
 CFG file 47
Max_Inlist_Values 124
MAX_INLIST_VALUES
 universe parameter 87
middleware
 checking availability on a machine 22
middleware, checking connectivity with network
 layer 24
Minus 124
MINUS 124

N

Native Int64 Available 82
Network layer
 checking available middleware 22
network layer, checking connectivity with
 middleware 24
network layers, listing supported 19
New Connection wizard
 accessing 28
New Connection wizard, accessing from
 Designer 27
NO_DISTINCT 124

O

ODBC
 exposing database layer in server mode 50
 Lookup parameters 50
ODBC Cursors 80
OLAP_Clause 125
OLAP_CLAUSE 125
Online Customer Support 153

Optimize Execute 66

Oracle

datatypes 143

OUTERJOINS_GENERATION 126

OVER_CLAUSE 129

overview

connection parameters 28

creating a connection 16

OWNER 129

P

parameter types, configuration 42

Password_Encryption 67

PATH_FINDER_OFF

universe parameter 87

personal

connection 30

ping, checking database access 25

PREFIX_SYS_TABLE 130

PrimaryKey Available 79

PRM file

BACK_QUOTE_SUPPORTED 112

CASE_SENSITIVE 112

CHECK_OWNER_STATE 112

CHECK_QUALIFIER_STATE 113

COMMA 113

CONCAT 114

DATABASE_DATE_FORMAT 115

DATATYPE_BLOB 115

DATATYPE_DOUBLE 115

DATATYPE_DTM 116

DATATYPE_INT 116

DATATYPE_NULL 116

DATATYPE_STRING 116

DATE_WITHOUT_QUOTE 117

edit parameters 109

EXT_JOIN 117

EXT_JOIN_INVERT 118

EXTERN_SORT_EXCLUDE_DISTINCT
118

GROUPBY_EXCLUDE_COMPLEX 119

PRM file (*continued*)

GROUPBY_WITH_ALIAS 119

GROUPBY_WITHOUT_CONSTANT 120

GROUPBYCOL 120, 121, 125, 130,
132, 134

INTERSECT 121

KEY_INFO_SUPPORTED 122

LEFT OUTER 122

LENMAXFORCOLUMNNAME 123

LENMAXFORTABLENAME 123

LENMAXFORVARCHAR 123

list of files 106

MINUS 124

NO_DISTINCT 124

OLAP_CLAUSE 125

OUTERJOINS_GENERATION 126

OVER_CLAUSE 129

OWNER 129

PREFIX_SYS_TABLE 130

QUALIFIER 131

QUOTE_OWNER 131

reference 106

reference list 111

REFRESH_COLUMNS_TYPE 132

REVERSE_TABLE_WEIGHT 132

RIGHT OUTER=\$(+) 133

RISQL_FUNCTIONS 133

SORT_BY_NO 135

TABLE_DELIMITER 120

UNICODE_PATTERN 135

UNION 135

USER_INPUT_DATE_FORMAT 136

USER_INPUT_NUMERIC_SEPARATOR
137

view parameters 109

Protocols

CFG file 50

Provider CLSID 82

Q

QUALIFIER 131

Index

Quote_Owner 131
QUOTE_OWNER 131

R

Red Brick
 datatypes 144
 TIME support in WHERE 145
 TIMESTAMP support in WHERE 145
Refresh_Columns_Type 132
REFRESH_COLUMNS_TYPE 132
REPLACE_COMMA_BY_CONCAT
 universe parameter 87
resources 150
REVERSE_TABLE_WEIGHT 132
RIGHT OUTER=\$(+) 133
RISQL_FUNCTIONS 133

S

SBO file 58, 59, 60, 68, 70, 72
 Array Bind Available 58
 Array Fetch Available 58
 Array Fetch Size 59
 configuring a JDBC connection 36
 Databases parameters 56
 Defaults parameters 56
 Description File 62
 editing 54
 Family 63
 Native Int64 Available 82
 ODBC Cursors 80
 Optimize Execute 66
 Password_Encryption 67
 Provider CLSID 82
 SQL Parameter File 71
 Strategies File 72
 structure 56
 Transaction Available 73
 Unicode 74
 V5toV6DriverName 76
 viewing 54

SBO files
 parameter categories 56
SBO files, installed 53
secured
 connection 30
SELFJOINS_IN_WHERE
 universe parameter 87
settings
 CFG parameter reference 44
shared
 connection 30
SHORTCUT_BEHAVIOR
 universe parameter 87
SORT_BY_NO 135
SQL External Extension
 CGF file 48
SQL parameter
 PRM file list 111
SQL Parameter Extension
 CFG file 48
SQL Parameter File 71
SQL Server
 datatypes 142
start
 creating a connection 28
stored procedures
 about JavaBean connections 38
Strategies Extension
 CFG file 48
Strategies File 72
support
 customer 153
 locations 152
 technical 153
 web site 153
supported database engines, listing 19
supported network layers, listing 19
Sybase
 datatypes 145
Sybase, connectivity check example 25
system architecture 11

T

TABLE_DELIMITER [120](#)
technical support [153](#)
Teradata
 datatypes [146](#)
THOROUGH_PARSE
 universe parameter [87](#)
training, on Business Objects products [153](#)
Transaction Available [73](#)
TRUST_CARDINALITIES
 universe parameter [87](#)

U

Unicode
 SBO file [74](#)
UNICODE_PATTERN [135](#)
UNICODE_STRINGS
 universe parameter [87](#)
UNION [135](#)

universe parameter
 FORCE_SORTED_LOV [114](#)
 reference list [87](#)
USER_INPUT_DATE_FORMAT [136](#)
USER_INPUT_NUMERIC_SEPARATOR [137](#)

V

V5toV6DriverName [76](#)
view
 PRM file parameters [109](#)

W

web
 customer support [153](#)
 getting documentation via [151](#)
 useful addresses [150](#)
web sites
 support [153](#)
 training [153](#)

Index