Business Objects

Crystal Reports

How to use SQLCON32 to diagnose ODBC driver issues

Overview

This document describes how to use Sqlcon32, a tool for testing Open DataBase Connectivity (ODBC) data source behavior. Each functional Sqlcon32 component is examined in this document. All Crystal Reports (CR) users, who use ODBC connections in their reports, will benefit from this document. For information about troubleshooting database connectivity problems in Crystal Reports XI, click here.

Contents

INTRODUCTION **2**
Where to find the Sqlcon32 utility *2*

CONNECTIVITY **2**
How Crystal Reports Connects to ODBC *3*
Connection types used by Crystal Reports and Sqlcon32 ..... *4*
Driver Connect 4
Once the data source has been selected, the associated ODBC driver will prompt for logon information. *5*
Simple Connect 5
Silent Connect 6

INCORRECT DATA OR REFRESH ERRORS **7**
Steps to test the SQL statement in Sqlcon32: 7

LISTING OF TABLES **8**

FIELDS OF A TABLE **9**

LISTING OF PROCEDURES **10**

PARAMETERS OF A PROCEDURE **11**
Parameter Types *13*

FIELDS OF A PROCEDURE **13**

FINDING MORE INFORMATION **15**

CONTACTING BUSINESS OBJECTS FOR TECHNICAL SUPPORT **15**

Introduction

Sqlcon32 is an ODBC connectivity tool for testing behavior with ODBC data sources. Sqlcon32 is a valuable tool in diagnosing and isolating problems within the three components of an ODBC connection:

- The Crystal Reports' database driver for connecting to ODBC, (**crdb_odbc.dll** in CR 9 and later; **p2sodbc.dll** in CR 8.5 and earlier)
- The ODBC layer, the ODBC driver that is being used as configured in the ODBC Data Source Administrator
- The database driver

Knowing **when** the error or the problem is occurring in Crystal Reports (CR) is critical as to how to test with Sqlcon32. With proper use, Sqlcon32 can diagnose and isolate most issues with an ODBC connection.

Where to find the Sqlcon32 utility

To find the Sqlcon32 utility, 32bit ODBC Connectivity & Query Test Utility, search for the filename, **sqlcon32.zip**, on our support site at http://support.businessobjects.com/search.

Connectivity

In some situations, issues will arise when connecting to the database through the ODBC driver, Crdb_odbc.dll. In Crystal Reports, a connection to the database is requested during the following scenarios:

When creating a new report.

When running an existing report (without the appropriate connection open).

When using the command, **File > Logon Server**.

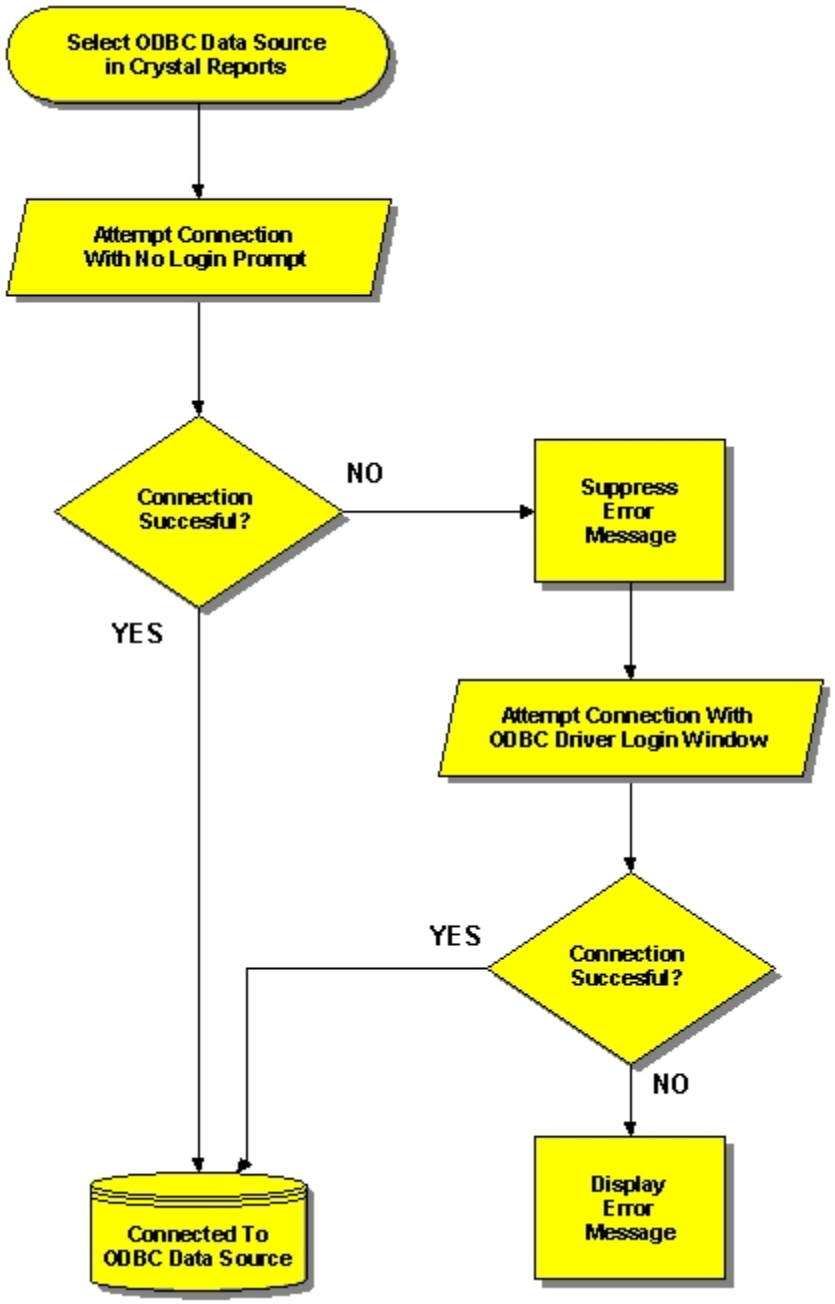
In the case of creating a new report, Crystal Reports is both **opening** a connection to the database and **retrieving** a list of table objects (tables, views, synonyms, system tables, and/or stored procedures).

To isolate if this is a connection issue in Crystal Reports, attempt to connect through the command, **File > Logon Server**. If an error occurs at this time, the problem is with the connection to the database. Since the **Logon Server** command is only attempting to **open** a connection to the database, you can isolate the issue to the database connection.

How Crystal Reports Connects to ODBC

Crystal Reports will make one or two attempts to connect to an ODBC data source. The first attempt suppresses any request for user input. If there is enough information in the ODBC data source to connect, the connection is automatically made. This can streamline the report creation process. However, if the initial connection attempt fails, Crystal Reports will make a second connection attempt, which will force a logon screen to appear.

The process of Crystal Reports attempts to connect to an ODBC data source is demonstrated in the following flowchart:



NOTE	Failure of the initial connection attempt will never produce an error in Crystal Reports. The second connection attempt is made immediately following the failed first connection attempt.

Connection types used by Crystal Reports and Sqlcon32

Sqlcon32 offers three methods of connecting to an ODBC data source to match the same three connection methods used by Crystal Reports:

- Driver Connect
- Simple Connect
- Silent Connect

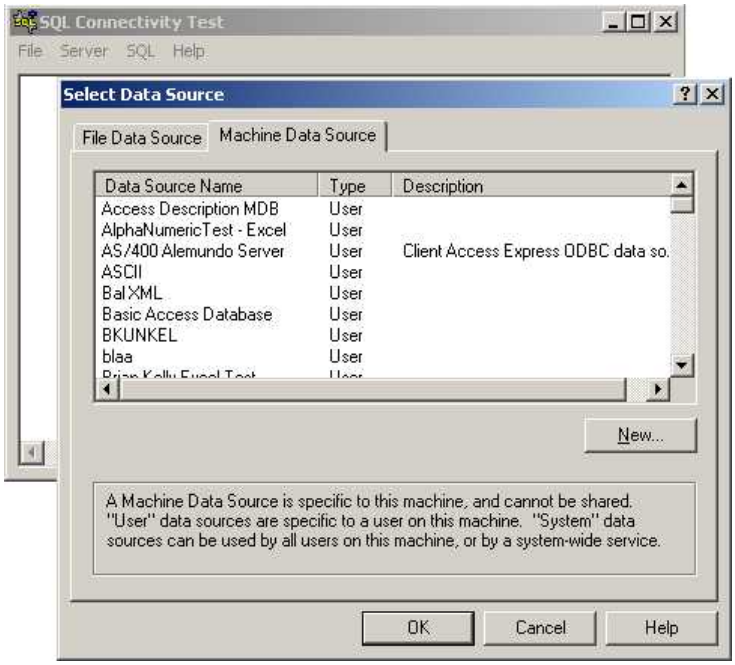
Driver Connect

Driver Connect prompts users to select an ODBC data source and to provide logon information using the ODBC driver's customized logon window. Testing with *Driver Connect* is useful to reproduce the connection made in Crystal Reports' second connection attempt, when needed (see *How Crystal Reports Connects to ODBC*)

TECH TALK	Crystal Reports passes the ODBC API call **SQLDriverConnect** to the ODBC driver to connect on a second attempt, if necessary. This is the same as the ODBC API call made when Sqlcon32 attempts a connection using *Driver Connect*.

Driver Connect is accessed from the **Server** command of Sqlcon32 and provides the most controlled method of connecting to ODBC data sources.

HINT	If a logon window appears when connecting in Crystal Reports, use *Driver Connect* to test in Sqlcon32.



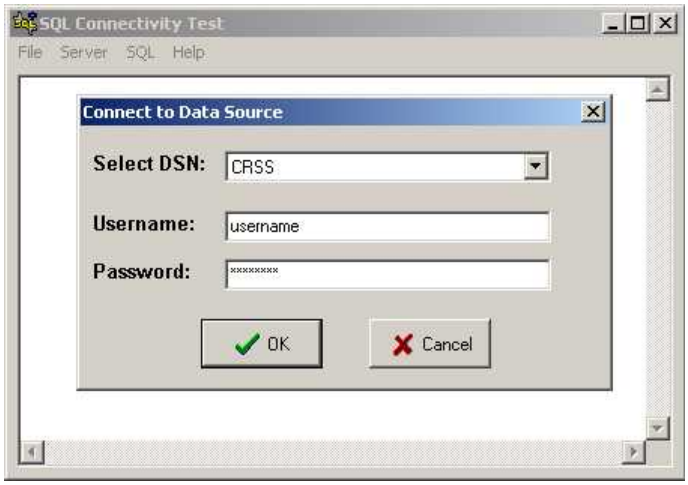
In the Select Data Source dialog box, data source options are listed under the *File Data Source* tab or the *Machine Data Source (System and User data sources)* tab.

NOTE	The Select Data Source dialog box above is generated by the ODBC Driver Manager.

Once the data source has been selected, the associated ODBC driver will prompt for logon information.

Simple Connect

Simple Connect prompts for an ODBC data source, username, and password using a common logon window. *Simple Connect* is accessed from the *Server* command of Sqlcon32.

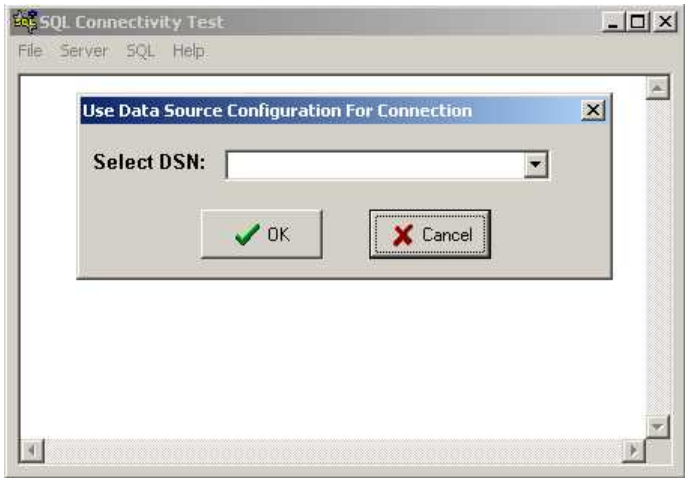


TECH TALK	*Simple Connect* uses the ODBC API call **SQLConnect**. **SQLConnect** is an alternate method of connecting to an ODBC data source. **SQLConnect** is commonly used for the first connection attempt to PC-type database ODBC data sources.

HINT	*Simple Connect* should be used to connect to ODBC data sources for PC-type database files. This reproduces how Crystal Reports connects to these ODBC data sources. In that case, *Username* and *Password* are generally left blank, unless security is placed upon the database.

Silent Connect

Silent Connect prompts for an ODBC data source and nothing else. *Silent Connect* is accessed from the **Server** command of Sqlcon32.



TECH TALK	*Silent Connect*, like *Driver Connect*, uses the ODBC API call, **SQLDriverConnect** to connect. However, in this case, **SQLDriverConnect** is set to not prompt for logon information.

Silent Connect assumes that there is enough information in the ODBC data source to make a connection. If the connection fails, a message appears recommending *Driver Connect* or *Simple Connect* as connection methods.

HINT	*Silent Connect* reproduces the initial connection attempt made in Crystal Reports to server/client based ODBC data sources. It is assumed that all logon information is available in the ODBC data source. If there is not, the connection attempt will fail.

Incorrect Data or Refresh Errors

When a report is created with an ODBC connection, a SQL statement is generated in the report and passed to the ODBC driver. If the report is suspected of displaying incorrect records, or if an error occurs when a report is refreshed, the SQL statement can be copied directly from the report and pasted into Sqlcon32 to compare if the same data is returned or if the same error occurs.

NOTE	If a record selection formula is used in the report, see if this formula was converted in whole or in part to the WHERE clause for the SQL. If only part of the record selection formula was converted to the WHERE clause, a higher number of records will appear in SQLCON32. This is because the Report Designer will perform additional record filtering once the record set is retrieved from the server.

Steps to test the SQL statement in Sqlcon32:

1. In the Report Designer, copy the SQL statement generated by Crystal Reports from the command **Database > Show SQL Query** window.

CAUTION	When working with stored procedures, Crystal Reports will show the *{call}* statement in the command **Database > Show SQL Query** window of the Report Designer. However, "?" characters are shown as placeholders for parameter values. When using this *{call}* statement in Sqlcon32, the "?" characters must be replaced with the actual parameter values. For example: {call YourDatabase.YourOwner.YourProcedure(?,?)} ->{call YourDatabase.YourOwner.YourProcedure(1,'hello world')}

2. After successfully connecting in Sqlcon32 to the same ODBC data source as used in your Crystal Reports, go to the command **SQL > Execute a Query**. Paste the SQL statement into the pop-up window and click **OK**. Data or an error message will appear in Sqlcon32's main window.
3. If incorrect data or an error message occurs in Sqlcon32 as well as the Report Designer, the problem resides with the ODBC driver or the database driver. The issue needs to be addressed by the manufacturer of the ODBC driver or the Database Administrator. If

the correct data occurs here, the problem may reside with Crystal Reports' ODBC driver, Crdb_odbc.dll.

CAUTION	Sqlcon32 converts all data to character data for display purposes, whereas Crystal Reports will preserve data typing for data so that data can be used for formatting and formulas. This should be kept in mind during testing of values with data types that are not character-based. Testing in Sqlcon32 may not be appropriate in that case.

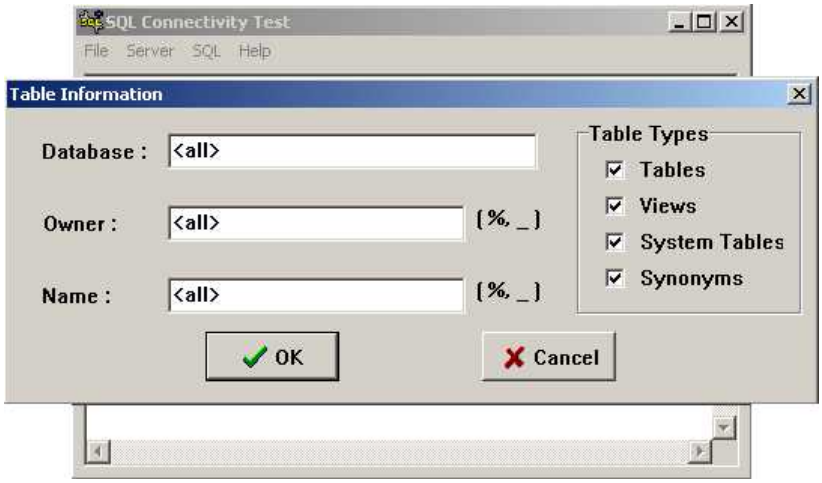
TECH TALK	When Sqlcon32 passes SQL to the ODBC driver, there is no structural information passed or retrieved about the fields. Sqlcon32 simply displays the data, returned by the database server, in text format without concerning itself with field data types. Crystal Reports needs to associate both data and data type structures to each field from the database.

Retrieving Structural Information

Listing of Tables

Sqlcon32 can retrieve a list of tables via the command **SQL > Get Listing of Tables** from an ODBC data source.

HINT	The **Get Listing of Tables** command in Sqlcon32 reproduces the action that Crystal Reports uses when a list of tables appears in the Data Explorer after connecting to an ODBC data source.



A **Table Information** dialog box appears, allowing for an option to make the table list more refined by specifying database, owner and table name information.

The "**Database:**" text box allows the user to type in a database qualifier, limiting the table list to a specified database from the server. For example, entering **Northwind** would retrieve all tables from the

Northwind database from Microsoft SQL Server. The default behavior is to include all databases.

The "**Owner:"** text box allows the user to type in an owner qualifier, limiting the table list to a specified owner for tables. For example, entering **CRYSTAL** would retrieve all tables created by user CRYSTAL on an Oracle server. The default behavior is to include all table owners.

The "**Name:**" text box allows the user to retrieve a subset of tables using wildcard characters. For example, typing in **ST**% would retrieve all tables beginning with "ST". The default behavior is to include all table names.

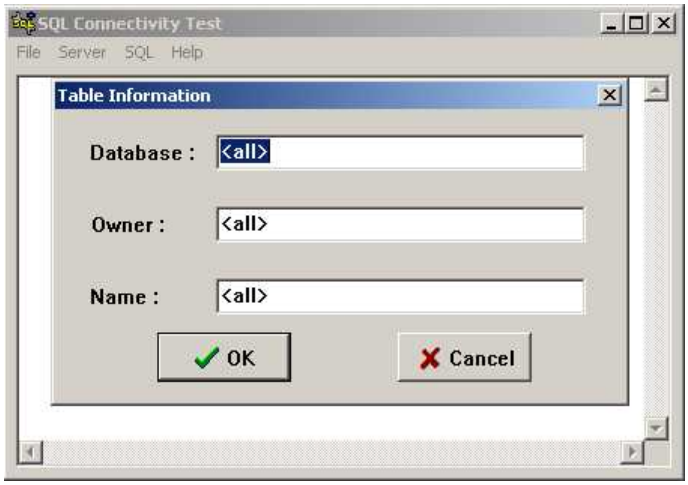
NOTE	The "%"wildcard character represents a search for one or more characters. The "_ " wildcard character is a placeholder for a single character within string. For example: "ST%" searches everything that starts with "ST" "ST_CK" searches everything that matches "ST_CK", like "STICK", "STACK", "STOCK", "STUCK"

Using the check box options below the Table Types heading can further refine the table lists. Removing or adding the checkmarks can retrieve any combination of Tables, Views, System Tables, and Synonyms.

Fields of a Table

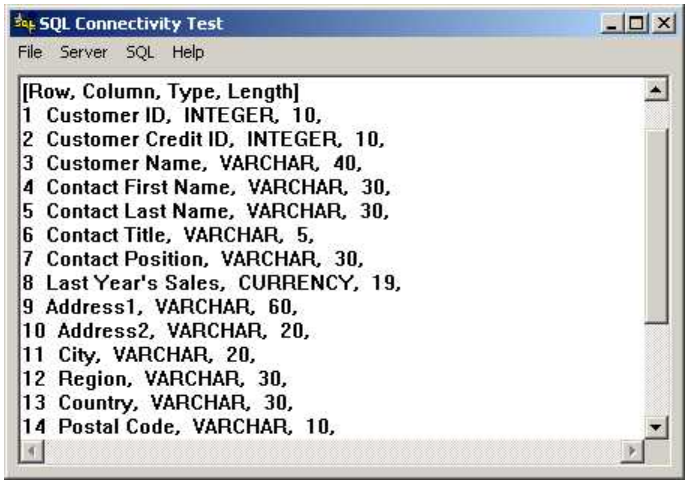
Sqlcon32 can retrieve a list of fields from a table via the command **SQL > Get Fields of a Table**.

HINT	The **Get Fields of a Table** command in Sqlcon32 reproduces the action in Crystal Reports when a table is added to a report. When a table is added to a report, field information is retrieved so that fields may be placed on the report.



NOTE	The Table Information dialog box cannot use wildcards, "%" or "_". If the symbol "(%,_)" is next to the input boxes, this indicates that these wildcards are available as a search option.

A **Table Information** dialog box appears, allowing for an option to make the field list more refined by specifying database, owner and table name. The default behavior is to include fields from all databases, owners and tables. After the **OK** button is clicked, field information appears in Sqlcon32's main window.



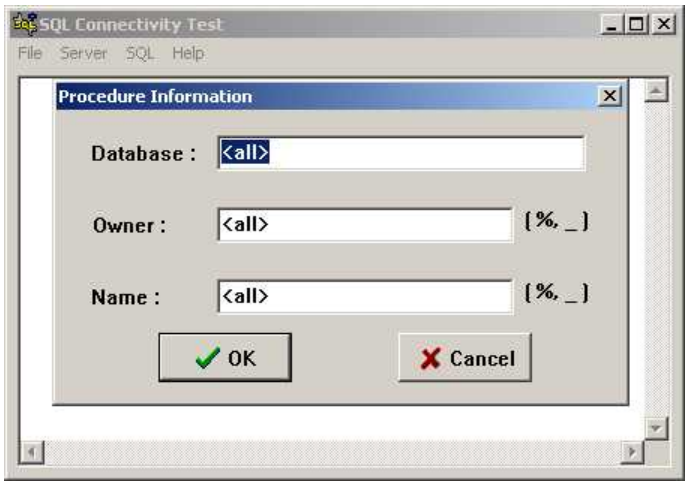
The row number, field name, field data type, and data type length for each field are displayed.

Listing of Procedures

Sqlcon32 can retrieve a list of tables from an ODBC data source via the command **SQL > Get Listing of Procedures** .

HINT	The **Get Listing of Procedures** command in Sqlcon32 reproduces the action in Crystal Reports when a list of stored procedures appears in the Data Explorer after connecting to

	an ODBC data source.



NOTE	The Procedure Information dialog box can use wildcards, "%" or "_". If the symbol "(%,_)" is next to the input boxes, this indicates that these wildcards are available as a search option.

A **Procedure Information** dialog box appears, allowing for an option to make the procedure list more refined by specifying database, owner and table name. The default behavior is to include fields from all databases, owners and tables.

After the **OK** button is clicked, field information appears in Sqlcon32's main window.

NOTE	The "%"wildcard character represents a search for one or more characters. The "_ " wildcard character is a placeholder for a single character within string.

Parameters of a Procedure

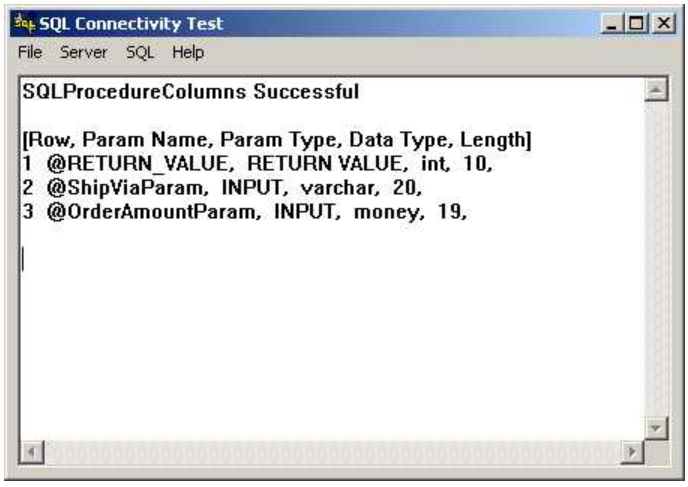
Sqlcon32 can retrieve a list of parameters from a stored procedure via the *SQL* > *Get Parameters of a Procedure* command.

HINT	The **Get Parameters of a Procedure** command in Sqlcon32 reproduces the action in Crystal Reports when a window prompts for parameter values after a procedure is added to a report.

NOTE	The Parameters of a Procedure Information dialog box cannot use wildcards, "%" or "_". If the symbol "(%,_)" is next to the input boxes, this indicates that these wildcards are available as a search option.

A **Procedure Information** dialog box appears, allowing for an option to make the parameter list more refined by specifying database, owner and table name. The default behavior is to include parameters from all databases, owners and tables.

After the **OK** button is clicked, parameter information appears in Sqlcon32's main window.



The row number, parameter name, parameter type, parameter data type, and data type length are displayed.

NOTE	The parameter type will return information on how the parameter functions.

Parameter Types

Sqlcon32 will show the parameter type of a stored procedure parameter through the command **SQL > Get Parameters of a Procedure** . Parameter types will fall into one of the following categories:

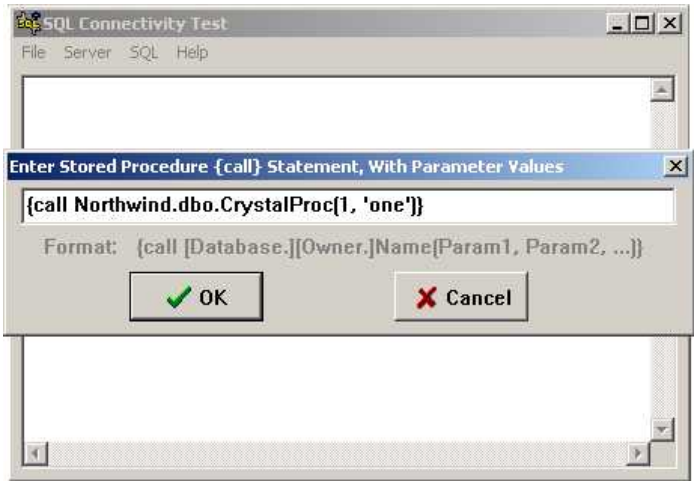
- RETURN VALUE represents the record set of the stored procedure. This is a standard parameter value from a SQL-type stored procedure. This parameter type is transparent to Crystal Reports.
- INPUT represents an input parameter from the stored procedure. This parameter is the most common parameter type. It requires data to be inserted into the parameter to be passed to the stored procedure for processing. Crystal Reports produces an *Enter Parameter Values* window for users to enter parameter values for input parameters.
- OUTPUT represents an output parameter from the stored procedure. Output parameters return single values, separate from the stored procedure record set. Output parameters are not supported by Crystal Reports.
- RESULT COLUMN represents a column of the stored procedure's record set. Some ODBC drivers will classify fields in a stored procedure as result columns.
- INPUT/OUTPUT represents an input/output (IN/OUT) parameter from a stored procedure. Input/Output parameters both receive values for stored procedure processing and return values from the stored procedure. In general, Crystal Reports does not support input/output parameters.
- UNKNOWN represents a parameter type that is not known by the ODBC driver.

Fields of a Procedure

Sqlcon32 can retrieve a list of fields from a stored procedure via the command **SQL > Get Fields of a Procedure**.

HINT	The **Get Fields of a Procedure** command in Sqlcon32 reproduces the action in Crystal Reports after parameter values have been entered (if required) when Crystal Reports selects a stored procedure to add to a report.

After parameter values have been entered, Crystal Reports will execute the stored procedure, using an ODBC *{call}* statement, to retrieve a list of fields for the report. Likewise, **Get Fields of a Procedure** using an ODBC *{call}* statement to retrieve a list of fields.



OR

The {call} statement format, shown in the prompting window, has the following syntax:

{call [Database.][Owner.]Name(Param1, Param2, …)}

/Where [ ] (square brackets) denote values that do not apply to every ODBC driver./

/Database is the database qualifier of the stored procedure. For example, this would be the database on the server where the stored procedure resides. The default behavior is to include all databases./

/Owner is the owner qualifier of the stored procedure. The Owner is the user who created the stored procedure. For example, if the user CRYSTAL logged in and created the stored procedure, the stored procedure is qualified with the owner, CRYSTAL. The default behavior is to include all procedure owners./

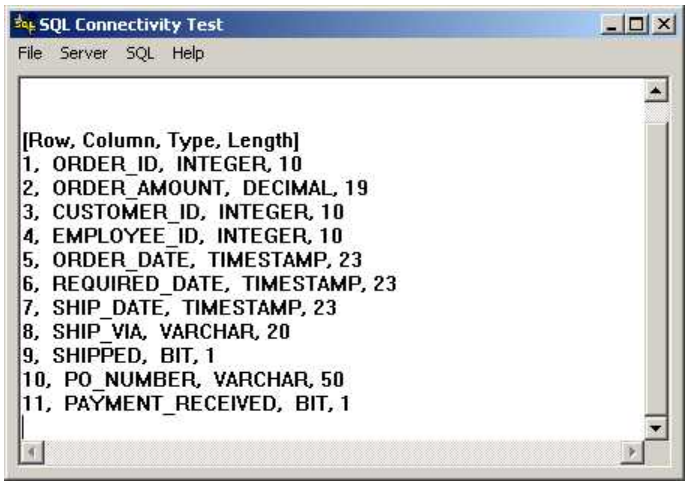
/Name is the name of the stored procedure. The default behavior is to include all procedure names./

The stored procedure values, if required, are listed in parentheses following the procedure name. If no input parameters exist for a stored procedure, the parentheses are dropped.

For example: {call Northwind.dbo.NoParamProc}

CAUTION	Crystal Reports shows the {call} statement in the **Database > Show SQL Query** command of the Report Designer. However, "?" characters are shown to denote parameter values. When using this {call} statement in Sqlcon32, the "?" characters must be replaced with the actual parameter values.

Once **OK** is clicked, field information appears in Sqlcon32's main window.



Finding More Information

The knowledge base article, **How to use SQLCON to test an ODBC Driver and Database Connectivity**, can be found on our support site at http://support.businessobjects.com/search. Search for article c2007753.

For information about troubleshooting database connectivity problems in Crystal Reports XI, click here.

Contacting Business Objects for Technical Support

We recommend that you refer to the product documentation and that you visit our Technical Support web site for more resources.

Self-serve Support:

http://support.businessobjects.com/

Email Support:

http://support.businessobjects.com/support/answers.asp

Telephone Support:

http://www.businessobjects.com/contact/support.asp

▶ www.businessobjects.com