

Seagate Crystal Reports 5 and 6

Performance Considerations for Record Selection Formulas

Overview

This document discusses how you can improve report performance by listing the issues you need to consider when creating record selection formulas in Crystal Report Designer (CR). Most importantly, it will cover how CR passes record selection formulas to the SQL query when reporting on SQL/ODBC data sources.

Contents

INTRODUCTION	2
PERFORMANCE CONSIDERATIONS	2
PARAMETER FIELD SELECTION - SQL QUERY - SPEED ISSUES	4
Problem	4
Solution	4
CONTACTING CRYSTAL DECISIONS FOR TECHNICAL SUPPORT	5

Introduction

Record selection, the task of paring down the data in your reports to include only the data required for your report, is a crucial step in report creation. You will rarely want a listing of all the information in a database. Most often you will be interested in only the sales in a given time period or for a certain product, etc. For example, a sales report may be designed to only include sales from one product line for the last calendar month.

Performance Considerations

There are a number of performance-related items that you should consider when you are setting up your selection requests:

- Record selection will be faster if it is based on indexed fields instead of non-indexed fields.
- If you have based record selection on indexed fields, make sure the Use Indexes or Server for Speed option is toggled on in the Report Options dialog box.
- Avoid performing record selection based on formula fields if at all possible because it will result in less efficient reporting. For example, assume you have a formula field (@ExtendedPrice) in your report that returns the extended price of a line item (Quantity * Price). If you base your selection criteria on that formula (@ExtendedPrice > 1000, for example), the SQL server will not understand the formula so the program will not pass the selection criteria down to the server. Instead it will retrieve all of the records from the server, and then it will apply record selection on the client machine. This can tie up network resources and slow processing time considerably.

- Try to avoid subscript ranges such as:

```
{file.FIELD}[1 to 5]
```

The program parses selection formulas and converts anything it can to SQL so that the bulk of the work can be off-loaded to the SQL server. Because there is no SQL equivalent to subscript ranges, SQL pass through will not occur. Subscripting of only the first character works, but subscripting multiple characters does not.

- When using SQL/ODBC data sources, if you are unsure if the record selection is passing through to SQL or not, check it by choosing the Show SQL Query command from the Database menu. If the SQL query does not have a WHERE statement or if the WHERE statement does not mention all of your fields that you are dealing with in your record selection, then you will need to work through the formula again since the translation did not occur properly.
- Do not perform any data type conversions in the record selection formula (for example, using the DTSToDateTime, DateTimeToDate, Totext() Functions). Such conversions cannot be translated to SQL so SQL pass through will not occur.

- With both data files and SQL databases, the program parses the entire selection formula and passes down whatever it parts of the criteria it is able to translate (pass), wherever they may physically appear in the formula.

Thus, if the formula finds criteria it can pass, then criteria that it cannot, then criteria that it can, it passes down the first part, skips the second, and then passes down the third.

In the case of data files, it passes down the criteria that it can to the database translation layer. In the case of SQL databases, it passes down to the server the criteria that it can in the form of a WHERE clause.

While there are exceptions, as a general rule the program can pass down any part of the record selection formula that compares a field with a constant. Typically, this means that it can pass down any kind of record selection criteria that can be set up in the Select Expert (field equal to, one of, less than, greater than, less than or equal, greater than or equal, between, starting with, or like constant).

There are two special selection formula situations that need to be considered: using AND and using OR. In these situations, there are multiple conditions in the record selection formula and some can be passed down while others cannot.

AND situations

```
{customer.REGION} = "CA" and  
{customer.CUSTOMER ID}[3 to 5] = "777")
```

In this situation, the program sees that it can pass down the condition before the AND operator but not the condition after. Since the only records that will meet the second condition will have to meet the first as well, it passes down the first condition, retrieves the data set that satisfies the condition, and then applies the second condition to only the retrieved data. The rule for AND situations is that the program passes down whatever conditions it can.

If all of the conditions in an AND situation can be satisfied on the server or in the database DLL, the program passes them all down.

OR situations

```
{customer.REGION} = "CA" or  
{customer.CUSTOMER ID}[3 to 5] = "777")
```

In this situation, the program also sees that it can pass down the condition before the OR operator but not the condition after. Since there are records that can satisfy the second condition without satisfying the first, passing the first condition down does not make any sense because it will retrieve an incomplete data set. In other words, even if it retrieves all the data that satisfies the first condition, it will still have to retrieve all the data in the table(s) so it can apply the second condition in Crystal Reports. Thus, instead of duplicating parts of the data retrieval, the program passes nothing down. It retrieves all the data and then runs both tests in Crystal Reports. The rule for OR situations is that the program passes down all the tests or none of the tests.

If all of the tests in an OR situation can be performed on the server or in the database DLL, the program passes them all down.

Parameter Field Selection - SQL Query - Speed Issues

Problem

A report was created on a SQL database (in this case Microsoft SQL Server). A record selection on parameter fields is created in Crystal Reports so that users will be prompted to enter the filtering condition that the report will be processed on.

Does Crystal Reports pass the Parameter Field Selection as part of the SQL QUERY that gets processed on the SERVER SIDE? If yes, why does the report take so long to be processed as it seems like Crystal Reports is actually applying the selection criteria LOCALLY ON THE MACHINE instead of on the SERVER?

Solution

As long as the Database | Show SQL Query statement has NOT been modified, Crystal Reports WILL send the selection based on the parameter fields over to the SERVER to entirely be processed. Even though Crystal Reports DOES send the parameter field selection to the server, this will NOT show up in the Database | Show SQL Query option as it happens behind the scenes.

If the Database | Show SQL Query statement HAS been modified in Crystal Reports, then the parameter field selection will happen on the LOCAL machine and NOT the server. To rectify this problem, in the Database | Show SQL Query option, the RESET button needs to be clicked on (which will reset the query back to what it was originally set to before a change was made to it). Then, the parameter field record selection will be processed on the SERVER.

If a report is based on a QUERY FILE (.QRY), then the Parameter Field record selection gets processed ENTIRELY on the LOCAL machine and NOT the server. The concept of a query file is that Crystal Reports will execute the query on the server and then a result set will be returned back to Crystal Reports from which further report processing (such as a parameter field record selection) will occur.

If BOTH a query file and a record selection are to be used, the record selection should be done ahead of time in the QUERY FILE if the filtering condition is to be applied on the SERVER side.

The OPTIMAL solution for users who are looking to have:

- Customized control over the SQL statement
- Passing of parameters to database for filtering to occur on the SERVER side is:

Create a stored procedure with parameters. This is by far the most efficient manner to control SQL statements, prompt users for record selection parameters, and process ENTIRELY on the database server. The only supported databases whose stored procedures will work with Crystal Reports 5.0 are: Microsoft SQL Server, Sybase System 10/11, Oracle 7.2 and 7.3, and Informix 7.22 32-bit.

Contacting Crystal Decisions 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.crystaldecisions.com/>

Email Support:

<http://support.crystaldecisions.com/support/answers.asp>

Telephone Support:

<http://www.crystaldecisions.com/contact/support.asp>