Overview

This paper is written for Crystal Reports 7 Report Designer (CR) and Seagate Info Report Designer (SI). However, it also applies to CR 5 and 6. Exceptions include features such as mapping, which are specific to CR 7 and SI only.

Described are the steps CR takes during the two-pass reporting model, evaluation time functions, and the relation between record selection and Crystal Reports’ first pass over the data, among other topics.

Contents

INTRODUCTION .............................................................................................................. 2
THE TWO-PASS REPORTING MODEL .......................................................................... 2
First Pre-pass & BeforeReadingRecords formulas .................................................. 2
First Pass & WhileReadingRecords formulas ......................................................... 2
Second pre-pass ........................................................................................................ 3
Second pass & WhilePrintingRecords formulas ..................................................... 3
SECTION EVALUATION WITHIN THE TWO-PASS REPORTING MODEL .......... 4
Subreport evaluation ................................................................................................... 4
Special Note on Shared and Stored variables ......................................................... 5
Page on Demand Architecture ............................................................................... 5
How two-pass reporting affects page headers and footers .................................... 5
SPECIFYING WHICH PASS A FORMULA OCCURS IN ............................................. 5
BeforeReadingRecords ............................................................................................ 6
WhileReadingRecords ............................................................................................. 6
WhilePrintingRecords ............................................................................................. 6
EvaluateAfter ............................................................................................................ 7
THE RELATIONSHIP BETWEEN RECORD SELECTION AND THE FIRST PASS ..... 8
Scenario 1 – Server-side record selection ............................................................... 8
Scenario 2: Record selection performed partially on server and on client. ............... 9
Scenario 3 – Client-side record selection ............................................................... 9
Second Pass Issues .................................................................................................. 10
CONTACTING CRYSTAL DECISIONS FOR TECHNICAL SUPPORT ................. 10
APPENDIX A ............................................................................................................... 11
Crystal Reports 7 Evaluation Time ........................................................................... 11
Introduction

When previewing a report, Crystal Reports (CR) reads and manipulates data twice, using what is known as a ‘pass’ of the data, before the report displays onscreen. This two-pass report-processing model enables you to perform more complex calculations in reports, such as percent of subtotals.

To take full-advantage of this two-pass model, you need to understand the order in which report elements are evaluated when previewing reports. Evaluation time is a concept that controls many behaviors in CR. For example, CR can create graphs, crosstabs, maps or running total fields only on formulas that have already been evaluated.

Understanding the order in which CR reads and manipulates data allows you, as the report designer, to better plan your report, as you will have knowledge of what elements can and cannot be used together to give the desired results. For example, formulas that evaluate at the end of the two-pass evaluation process (such formulas are commonly known as ‘Print Time’ or WhilePrintingRecords formulas) cannot be used when creating maps, graphs, or crosstabs.

We strongly recommend that in addition to this document, you also review Appendix A of the Crystal Reports 7 - Users Manual, which outlines the multi-pass reporting concept in a step-by-step manner.

The Two-Pass Reporting Model

A pass refers to the process of reading records from the database and then manipulating them according to how you have designed the report. CR performs two main passes of a report before it previews (that is, prints the report to the screen):

- In the first pass, CR reads the data, record by record, from the database and creates basic summaries. These summaries are stored into a temporary file.
- In the second pass, CR reads the temporary file and performs more complex manipulations on the saved data

First Pre-pass & BeforeReadingRecords formulas

Before making the first pass through the database, CR evaluates any flat formulas in the report. Flat formulas are formulas that do not reference any database fields, for example:

2 * 50

First Pass & WhileReadingRecords formulas

CR reads the records from the database during the first pass. During the first pass CR evaluates any recurring formulas for each record. Recurring formulas are formulas that reference database fields for the current record only.

Referring to records other than the current record (for example, by using the Previous or Next functions), or including summary functions forces a formula to occur during the second pass.
The entire first pass process occurs for each individual record as CR retrieves it from the database, and saves the results (grouping, summaries, formula values, and so on) in a TMP file.

Figure 1. Diagram of the first pass

Second pre-pass
Group and TopN or BottomN sorting occurs here. The actions are based on the summaries created during the first pass. No additional records are read during the pre-pass.

Second pass & WhilePrintingRecords formulas
This pass is where more complex manipulation of the records occurs. At this point, CR reads the TMP file containing the saved records from the first pass, instead of reading directly from the database a second time.

*WhilePrintingRecords* refers to the stage within the second pass when CR evaluates formulas that reference more than one record. As discussed previously, including certain functions in a formula automatically forces the formula to evaluate *WhilePrintingRecords* because the function requires all records to have been read before it can calculate correctly.

Examples of second pass functions:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Any Inserted summary field</td>
<td>DistinctCount</td>
</tr>
<tr>
<td>WhilePrintingRecords</td>
<td>TotalPageCount</td>
</tr>
<tr>
<td>Next</td>
<td>RecordNumber</td>
</tr>
<tr>
<td>NextIsNull</td>
<td>GroupNumber</td>
</tr>
<tr>
<td>Previous</td>
<td>PageNumber</td>
</tr>
<tr>
<td>PreviousIsNull</td>
<td>TotalPageCount</td>
</tr>
<tr>
<td>Average</td>
<td>PageNofM</td>
</tr>
<tr>
<td>Sum</td>
<td>OnFirstRecord</td>
</tr>
<tr>
<td>Count</td>
<td>OnLastRecord</td>
</tr>
</tbody>
</table>

A rule of thumb is to determine whether you can apply a function to a single record, or whether more than one record is required for the function to calculate. If the answer is more than one record, this function will occur during the *WhilePrintingRecords* stage.
These second pass steps are evaluated for each section in the report, except for subreports and page formatting. Instead, these occur as you scroll through each new page. This is known as page-on-demand architecture.

Section Evaluation within the Two-Pass Reporting Model

In addition to the two-pass reporting concept, Crystal Reports (CR) is described as a banded report writer, meaning that CR writes the report vertically. The sections are evaluated in this order:

1. Report Header
2. Page Header
3. Group Header
4. Details
5. Group Footers
6. Report Footer
7. Page Footer

The second pass generally occurs for each section, primarily referring to graphing/mapping, running total fields, WhilePrintingRecords formulas and subreports. Objects are usually read from the top of the section to the bottom and then left to right.

However, when two WhilePrintingRecords formulas are inserted into a single section, one above another, the first formula inserted will be the first evaluated. If you later reverse the order by moving the formulas around, the evaluation will not change; they have been locked into their respective evaluation time positions. To break this, you will need to remove the formulas from the report and re-insert in them in the new order.

An example of how this banded, top-down report structure can impact reports is conditional formatting. You can perform conditional formatting on the Details section, based on the value of a formula in the Group Header; CR will have already determined the value of the formula in the Group Header by the time it reaches the Details section. However, you cannot conditionally format the Details section based on the same formula if that formula is in the Group Footer; while still in the Details section, CR would not yet know what is contained in the Group Footer.

Subreport evaluation

Subreports are one of the last objects to be evaluated in the second pass for each section. When CR encounters the subreport object, it generates this "report
within a report” with its own 2-pass structure. The main report then continues and completes the final formatting of the report.

Special Note on Shared and Stored variables

Evaluation time and subreport evaluation are especially important when using shared variables (or store and fetch functions in earlier versions of CR) to pass information from a subreport to the main report. The formula in the main report, which retrieves the values that were stored in the subreport, should be placed in a separate section that is beneath the section containing that subreport.

This is because the subreport will not have been recognized at the time that the formula is executed. Instead, the formula must be placed in the section that follows for example, group header A for the subreport and B for the formula. Once the subreport has finished and group header A is completed, the next section is evaluated in the second pass and the formula is executed.

Page on Demand Architecture

Page formatting, like subreports, is not generated for the entire report immediately upon clicking the Preview button.

For example, previewing a 20-page report will initially show only page 1. If there are subreports on the other pages they have not been generated and neither has the final formatting for those pages. As you scroll through the report, the subreports and the formatting are executed only as you come across them. This is known as “page on a demand architecture” and it is used to minimize processing time requirements.

You can change this ‘page on demand architecture’ by using the TotalPageCount function somewhere in the report. In order to determine the number of pages in the report, all subreports and general formatting must be performed in order for the total number of pages to be accurately determined.

How two-pass reporting affects page headers and footers

While you can conditionally suppress the contents of the page header and footer sections, you cannot conditionally suppress the space allotted for the sections.

CR reserves space for Page Header and Page Footer sections. If the conditional suppression criterion is met, the contents of a section do not appear, although the physical space that it would have occupied on the report still appears. This is because in the evaluation process, the page footers and headers are created before any section formatting is done.

Specifying which pass a formula occurs in

There is a set of functions in the formula editor in the Evaluation Time folder. Each function forces a specified formula to be evaluated at a certain time. These functions are:

- BeforeReadingRecords;
- WhileReadingRecords;
• WhilePrintingRecords;
• EvaluateAfter();

To ensure that a formula is evaluated at the correct stage in the evaluation process, we recommended including the appropriate evaluation time function at the beginning of the formula, followed by a semi-colon.

Without such a declaration, there may be uncertainty as to when the formula is evaluated in the evaluation process.

The following sections provide more detail about each of the Evaluation Time functions.

### BeforeReadingRecords

The BeforeReadingRecords function is rarely used since it applies to flat formulas only. There are a few circumstances that require this function. For example, if you enter a numeric parameter that is multiplied by a given value, the BeforeReadingRecords function is used:

```plaintext
@Flat formula
{?NumericParam} * 1.45
```

### WhileReadingRecords

The WhileReadingRecords function forces CR to evaluate a formula during the first pass. If the formula contains a function that must occur in the second pass, CR’s Formula Editor displays the error message “This function cannot be used because it must be evaluated later”. In this case, you can either:

• add WhilePrintingRecords to the beginning of the formula
  or
• remove the second pass function from the formula

### WhilePrintingRecords

You can force CR to evaluate any formula during the second pass by using the WhilePrintingRecords function. However, once the WhilePrintingRecords function is included in a formula, it has all of the limitations of the second pass formulas. For example, you cannot create charts or maps on second pass formulas because CR evaluates the former before it evaluates the latter during the second pass.

Generally speaking, WhilePrintingRecords should be the default for most formulas, unless you intend to insert summaries on that field.

An example of when to use the WhilePrintingRecords function is when declaring variables that are to be used in more than one formula on the report, in and in different sections. This is a very common issue when calculating running totals using formulas, rather than using the Running Total Expert.

#### Incorrect example

The formula below is created and placed in the details section of the report:

```plaintext
@SetVariable
```
//increments variable X by the value of {Table.Field}
NumberVar X;
X := X + {Table.Field}

@SetVariable is meant to display a total that accumulates for each record.
While the formula @SetVariable will return a value for each record, it will
not remember that value when CR moves on to evaluate @SetVariable for the
next record in the report. In other words, the variable X will not accumulate:

<table>
<thead>
<tr>
<th>Record</th>
<th>Table.Field</th>
<th>@SetVariable</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>15</td>
</tr>
</tbody>
</table>

Since there is neither second-pass functions nor any summaries in this formula,
CR considers it a recurring formula and evaluates it during the first pass on a
record-by-record basis. However, to accumulate values, all the records need to
be first read by CR; in other words, the formula must be made into a
WhilePrintingRecords formula:

Correct example
//increments variable X by the value of {Table.Field}
WhilePrintingRecords;
NumberVar X;
X := X + {Table.Field}
The formula now returns the cumulative value of the variable X on each detail
line of the report:

<table>
<thead>
<tr>
<th>Record</th>
<th>Table.Field</th>
<th>@SetVariable</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>35</td>
</tr>
</tbody>
</table>

EvaluateAfter

This function forces CR to evaluate one formula after another formula. This is
used when it is critical for the calculation of one formula depends on the results
of another formula.

@SetVariable
NumberVar X;
X := {table.field} * {table.field2}

@UseVariable
EvaluateAfter({@SetVariable});
NumberVar X;
X = {table.field3}

**CAUTION**
If you have a series of formulas and @Formula2 is in the report footer you may then have another formula that occurs earlier and has:
EvaluateAfter (@Formula2)
If this happens, @Formula2 will be forced to evaluate twice thus potentially causing incorrect results.
If you do have formulas that need to occur sequentially within a single section, it is strongly recommended that the first formula have the appropriate evaluation Time set for it. The formulas that follow should use the EvaluateAfter function to sequence the formulas so that they occur in the correct order.

The Relationship between Record Selection and the First Pass

From a report processing perspective, the ideal situation is where CR can parse the entire record selection formula for a report into SQL so it can be passed to the database when the report is previewed. This enables a report’s record selection to occur on the server, so CR ends up reading only the records that need to be included in the report. By reducing the number of records read by CR, server-side processing speeds up the time it takes to preview a report.

There are three possible ways CR can handle record selection formulas:

- Translates the entire formula and passes it to the database so the record selection actually occurs on the database (known as server-side processing).
- Translates part of the selection formula and passes it to the database for record selection, while the remainder of the selection formula is performed by CR (known as client-side processing)
- Does not translate any of the formula, so the entire selection process occurs on the client side.

**NOTE**
For more detail on how record selection formulas affect report-processing speed, you can download the technical brief called *Performance Considerations when Creating Record Selection Formulas* (SCR_RecSel.pdf) from the Library area of [http://support.crystaldecisions.com](http://support.crystaldecisions.com).

The following three examples illustrate how each type of record selection formula, as described above determine the steps CR must take while reading the database during the First Pass.

**Scenario 1 – Server-side record selection**
Record selection formula is `{table.field} = 1000`

1. Flat formulas are evaluated
2. Record selection is translated for the database (SQL statement for ODBC/SQL databases, to native language for other database types. “Group by” on the server is also included here if this option is enabled). Request is sent.
3. WhileReadingRecords formulas occur
4. Grouping, Sorting and Inserted totals are created
5. Values added to crosstab object
6. Values added to temp file
CR returns to step 1 until all records have been read.

Scenario 2: Record selection performed partially on server and on client

Record selection is \{table.field2\} = 1000 and \{table.field\} = @FirstPassFormula

1. Flat formulas are evaluated
2. Portion of record selection that can be translated (\{table.field2\} = 1000) for the database is translated (SQL statement for ODBC/SQL databases, to native language for other database types. “Group by” on the server is possibly included here if this option is enabled). Request is sent.
3. WhileReadingRecords formulas occur
4. Remainder of record selection formula is performed only after @FirstPassFormula has been processed.
5. Grouping, sorting and Inserted summaries are created
6. Values added to crosstab object
7. Values added to temp file
CR returns to step 1 until all records have been read.

Scenario 3 – Client-side record selection

Record selection is \{table.field\} = @FirstPassFormula

1. Flat formulas are evaluated
2. ALL records are retrieved from database
3. WhileReadingRecords formulas occur
4. Record selection formula is performed only after @FirstPassFormula has been processed.
5. Grouping, sorting and inserted summaries are created
6. Values added to crosstab object
7. Values added to temp file
CR returns to step 1 until all records have been read.

Referring to a WhileReadingRecords formula as in a record selection formula causes CR to request all (as in a scenario # 3) or a large portion (as in scenario #2) of the database to be returned. This can slow performance tremendously. For this reason, we strongly recommended not referring to any formulas within the record selection. Even in regular formulas (that is, non-record selection formulas), referring to other formulas is not recommended due to increased to processing time requirements.
Notice also that in all 3 examples, inserted summaries, grouping and sorting all occur after *WhileReadingRecords* formulas. This explains why you can insert group summaries and grand totals on *WhileReadingRecords* formulas, but not on *WhilePrintingRecords* formulas. For the same reason, you cannot group or perform sorts on *WhilePrintingRecords* formulas.

The alternative to inserting a summary or grand total on *WhilePrintingRecords* formulas is to create a running total.

For more information on running totals, go to [http://support.crystaldecisions.com/docs](http://support.crystaldecisions.com/docs) and download SCR_RunningTotal.zip, which is a document on *Everything You Need to Know about Running Totals*.

**Second Pass Issues**

Because they occur prior to the *WhilePrintingRecords* stage of the second pass, the following features cannot be performed on second pass formulas (Refer to Figure 2):

- Mapping
- Running total fields
- Grouping
- Sorting
- Inserted summaries

The alternative in such cases is to perform the calculation on the database side using, for example, stored procedures. These calculations are then brought in as simple fields and any of the above procedures can be carried out.

**Contacting Crystal Decisions for Technical Support**

Along with this document, and the *Crystal Reports User’s Guide*, we recommend that you visit our Technical Support web site for further resources and sample files. For further assistance, visit us at the web sites below.

Technical Support web site:

[http://support.crystaldecisions.com/homepage/](http://support.crystaldecisions.com/homepage/)

Answers By Email Support:

[http://support.crystaldecisions.com/support/answers.asp](http://support.crystaldecisions.com/support/answers.asp)

Phone Support:

Tel: (604) 669-8379
Appendix A

Crystal Reports 7 Evaluation Time

The purpose of this appendix is to keep the diagram together to make it easier to view the entire Evaluation Time process in CR 7.

Figure 1. Diagram of the First Pass, CR version 7

The First Pass occurs in its entirety for each record and then starts again

Figure 2. Diagram of the Second Pass, CR version 7

The Second Pass occurs for each section with Subreports and formatting. Page on demand of pages occurring as each new page is scrolled ahead.