

Crystal Reports 9 - Repository

Using the Repository for More Productive Report Development

Overview

Improving the productivity of the report design process is one of the primary objectives for Crystal Reports 9. To accomplish this, a centrally managed location is required so that report designers can access, update, reuse and share report objects. To extend this concept, report objects can also be shared and reused among multiple report designers for even greater efficiency.

A common source for report components is now available with the Crystal Reports 9 Repository. These reusable report objects are stored in a central database, making the process of maintaining them very easy. One change can be easily propagated across any number of .RPT files.

The purpose of this white paper is to outline how you can use the Repository for more productive report design and maintenance.

Contents

WHAT IS THE CRYSTAL REPOSITORY?	2
BACKGROUND: REUSE OF REPORT OBJECTS PRIOR TO VERSION 9	2
IMPROVE PRODUCTIVITY BY SHARING REPORT OBJECTS WITH THE CRYSTAL REPOSITORY	2
<i>Benefits of Using the Repository – A Real-World Example</i>	2
UNDERSTANDING HOW THE REPOSITORY WORKS	3
ORGANIZING THE REPOSITORY	3
<i>Hosting Objects in the Repository</i>	3
EDITING REPOSITORY OBJECTS.....	6
<i>Disconnecting a Repository Object</i>	6
<i>Reconnecting an Updated Object to the Repository</i>	6
SHARING YOUR REPOSITORY WITH OTHER DESIGNERS	6
SHARING CONNECTED REPORT OBJECTS WITH OTHERS - EVEN OFFLINE	7
SUMMARY	7

What is the Crystal Repository?

The Crystal Repository is a database that contains common report objects that can be shared and reused. The types of report objects that can be shared via the repository are:

- Text Objects – Reusable text, such as company addresses or confidentiality statements
- Images – Pictures or logos
- Custom Functions – Business logic that can be reused by passing in new fields as variables
- SQL Commands – Encapsulated database commands that produce tables to report from. These SQL statements can also be parameter driven.

Background: Reuse of Report Objects Prior to Version 9

In the past, many report designers would keep a set of reports that contained pieces of reports that they would use over and over. In many cases, these reports would contain corporate logos, commonly used objects (like a company's address), and even some typical formula logic. Where previous versions of Crystal Reports focused on powerful reporting, Crystal Reports 9's focus is on productivity.

Report designers would copy and paste these objects between these "warehouse reports" and the newly created reports on a regular basis. This would save the designer some time, as they would not have to recreate these objects for each new report. But, what happens when business logic changes and needs to be updated? This happens quite often and is a critical piece of the reporting puzzle. The only tried-and-true method of the past didn't solve the problem of how to update business logic or formulas in reports that had already been created. The report designer would have to remember which reports may have been affected, open each individual report, and update them by hand-coding the changes. This process was cumbersome and time-consuming but mission critical. One common location to store and update report objects would dramatically improve this situation.

Improve Productivity by Sharing Report Objects with the Crystal Repository

When designing the Crystal Repository, all of the sharing techniques designers used were carefully studied. The new Repository is independent from all other databases connected to any report. It manages logic, formatting, and objects. But most importantly, by its very nature as a separate library (or database), it is referenced by the report when the report is opened, and, as a result, it can actually update the repository objects automatically for the report designer so that the added overhead of copying new logic and objects is eliminated.

Benefits of Using the Repository – A Real-World Example

Before Crystal Decisions, our company was called Seagate Software. All the sample reports that shipped with older versions of Crystal Reports contained contact information in the footer that mentioned Seagate Software in two text objects – one for copyright information and one for feedback.

To get all of these reports to use the repository, the original objects were removed from each report and were replaced with the corresponding repository objects and then saved. A few months later, the company name changed, and all the sample reports needed updating again. The person in charge of maintaining the samples, simply opened ONE of the reports, disconnected the two objects, containing "Seagate Software", from the repository, updated them to "Crystal Decisions", then dragged the newly updated fields back the repository. In that one step, the objects in the report were reconnected to the repository AND the update was saved back to the repository for all other reports to access.

Let's look at how the productivity numbers worked when we actually did use Crystal Reports 9 to update the sample reports that ship on the CD. Our report designer expert had to update 40 sample reports - twice - in 9 different languages. She actually had to do the updates twice as we've gone through 2 different logos since we started building Version 9. Before Crystal Reports 9, this would have taken 10 minutes per report:

10 minutes multiplied by 360 reports (40 reports X 9 languages) = 3600 minutes

3600 / 60 minutes per hour = 60 hours / 8 hours per day = 7.5 full days of her time just to make one of the logo changes

X 2 changes = **15 days** of the report design expert's time to make simple logo changes to 360 reports.

Using Crystal Reports 9 with the Crystal Repository, these same updates took just 10 minutes per language. Therefore only 90 minutes were needed to update all languages. This equates to an 8000% increase in efficiency over previous maintenance or a 99% reduction in hours worked.

Understanding How the Repository Works

In Crystal Reports 9, the Crystal Repository is a standalone database. By default, it is a Microsoft Access database that Crystal Reports maintains. This means that Crystal Reports internally understands how to create the tables and structures it requires to make the repository work. Remember that the Access database is just a sample. You can use any ODBC data source for your Crystal Repository, though Crystal Decisions officially supports only the following:

- Microsoft Access
- Microsoft SQL Server
- Oracle
- IBM DB2

Note	For information on specific requirements for these databases, refer to knowledge base article c2012767 on our support site at: http://support.crystaldecisions.com/kbase
-------------	---

To add an object to the Crystal Repository, it's as simple as dragging and dropping the object into a folder in the Repository Explorer.

Organizing the Repository

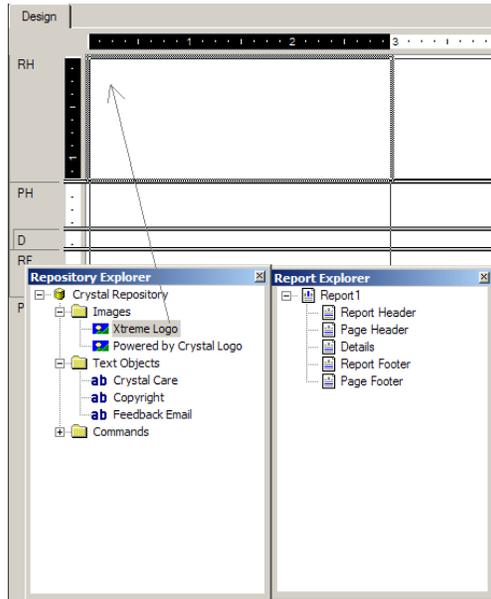
The Repository Explorer represents the repository database as a tree structure made up of folders and objects. The structure of the repository folders is up to the person working with the repository.

Hosting Objects in the Repository

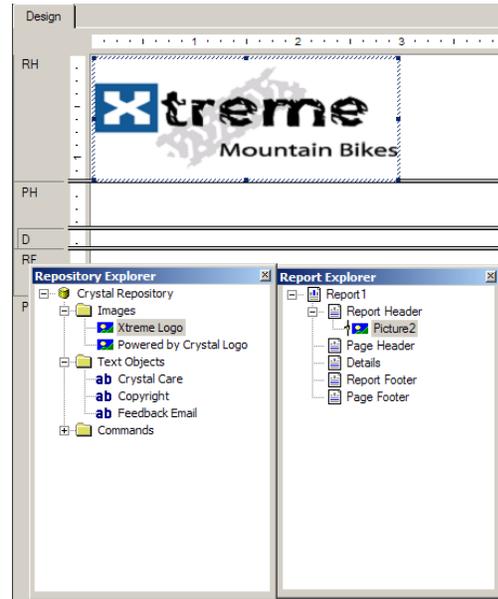
Within the Repository Explorer, the report designer has access to the repository objects that can be placed on the report design surface. Text objects and image objects are visible from the Repository

Explorer. To use Repository Objects in your reports, simply drag-and-drop these objects on to the report as required.

During Insert (Drag/Drop):



After Insert is complete:



SQL commands and their properties are visible in the Repository Explorer. You do not drag-and-drop these objects on the report as they are used in the Database Expert. You can use SQL Commands like database tables when you're designing your report. You can save your SQL statements in the repository for later use with the SQL Command in Repository feature.

While not visible in the Repository Explorer, there is one more report object type that is available for sharing. This type is a Custom Function. This new object type for Crystal Reports 9 can house business logic for reuse and portability. For example, handling typical numeric formatting for thousands can be created once and used as many times as needed. The following code is an example of a Custom Function available in the sample Crystal Repository that ships with Crystal Reports 9:

```
Function cdFormatNumberUsingScaling (value As Number, _
    nDecimalPlaces As Number, _
    thousandsSymbol as String, _
    millionsSymbol as String)

    nDecimalPlaces = Round (nDecimalPlaces)
    If nDecimalPlaces < 0 Or nDecimalPlaces > 9 Then
        nDecimalPlaces = 2
    End If
    If Abs (value) >= 1000000 Then
        cdFormatNumberUsingScaling = CStr (value / 1000000, nDecimalPlaces)+millionsSymbol
    ElseIf Abs (value) >= 1000 Then
        cdFormatNumberUsingScaling = CStr (value / 1000, nDecimalPlaces)+thousandsSymbol
    Else
        cdFormatNumberUsingScaling = CStr (value, nDecimalPlaces)
    End If
End Function
```

Editing Repository Objects

As with any report object, from time to time, Repository Objects will need to be updated. For example, the company address would need to be updated when the company changes office space. There are two steps required to update a repository object from the report.

- Disconnect the object from the repository and then make the change to the object as you would any other report object.
- Update the object in the repository.

Disconnecting a Repository Object

Because the Repository is basically a library to store objects, it has no update functionality in itself. This means that updates to objects must be done inside a Crystal Report. All connected repository objects are automatically considered read-only and cannot be changed. To update a specific object, you first need to disconnect it from the repository. This prevents inadvertent changes to objects. Disconnecting can be done by simply right clicking on the object and choosing "Disconnect" from the context menu.

After the object is disconnected, changes can be made as needed. These changes only affect the instance of the report object within that report. Remember, that the changes do not become committed to the repository and all other reports using the object until the object is connected back to the repository.

Reconnecting an Updated Object to the Repository

To reconnect a text object or image to the repository, simply drag and drop the object from the report back to its original repository object name/location in the Repository Explorer. A dialog box will appear to confirm if you want to update the original object or add a new one. It's that simple!

Sharing Your Repository with Other Designers

Can you share information between multiple report designers?

By creating the repository as a database, sharing can be realized in two ways:

- To share knowledge between reports
- To share that same knowledge with other report designers.

Crystal Reports provides the sample Crystal Repository in a small compact Microsoft Access database.

If you want to leverage the Repository for component sharing across designers, you can take advantage of Crystal's ODBC connectivity. This allows you to take advantage of the existing security of the database etc. Since Crystal Reports communicates with the repository through ODBC, you can point virtually any data source to a blank database and Crystal could turn it into its repository.

If you are trying to share the repository between multiple individuals, it's highly recommend that you use a client-server database that can be maintained and secured externally from Crystal such as Microsoft SQL Server, IBM DB2, Oracle and others. One of the key advantages to using a client-server database in place of the sample that Crystal Reports provides is that you can take the existing security for your users and continue it into the repository. If a user does not have write permission to a given database, they also will not have the ability to modify objects in the repository.

For example, if a report designer has access to Microsoft's SQL Server database, he could publish his repository to this database source (by importing his existing MDB) and give access to the repository to other colleagues. By sharing one repository, all the report designers will increase consistency across their reports and be more productive with their design efforts.

To inform the Crystal Reports application about the change to the location of the data source, just point Crystal Reports to the new data source and then load up the designer again. It's that simple!

CAUTION	If the new database requires a username or password to connect to it, these credentials will be prompted for upon initially opening the repository. But, opening and closing the repository explorer will not log off the server. Therefore, prompting is kept to a minimum.
----------------	--

TIP	Because the repository is now shared with others, it can be updated and maintained by others as well. If there is a need to limit the updating ability, this must be done on the database side by limiting the update rights of a particular user.
------------	--

Sharing Connected Report Objects with Others - Even Offline

What happens if the report needs to be sent to someone who does not have access to my repository?

Reports that have been connected to the repository are still portable. They do not need to be physically connected to a given repository to be viewable by others. Crystal Reports files were designed to save all objects within them so that if the repository is not available to check for updated versions of the objects, the report will load with the last version of the objects that were stored within the RPT file. This allows report designers to share RPT files without having to share the same repository. When the original RPT file is returned (with changes made by the second report designer) and opened, the repository will update the report as necessary.

Summary

Using the Crystal Repository as an integral part of sharing between reports and report designers, your report designers' productivity and efficiency will increase. Either on its own or in combination with other key features of Crystal Reports 9, your report designers will be improving their output for you and your organization.

The information contained in this document represents the best current view of Crystal Decisions on the issues discussed as of the date of publication, but should not be interpreted to be a commitment on the part of Crystal Decisions or a guarantee as to the accuracy of any information presented.

This document is for informational purposes only and is not intended as a guarantee or warranty of any kind. The information is provided AS IS. CRYSTAL DECISIONS MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT. Crystal Decisions will not be liable for any damages (direct, indirect or consequential) to you for your consumption, use or reliance on the information provided herein.

© Copyright 2002 Crystal Decisions, Inc. All rights reserved. Crystal Reports, Crystal Enterprise, and Crystal Decisions are the trademarks or registered trademarks of Crystal Decisions, Inc. All other trademarks referenced are the property of their respective owners.

Specifications and product offerings subject to change without notice.