crystal decisions
A SEAGATE COMPANY

Crystal Reports ASP Reporting

Troubleshooting the error: "Server Has Not Yet Been Opened - 2147192184"

Overview

This whitepaper discusses the causes and solutions of the error message “Server Has Not Yet Been Opened -2147192184” an error that is encountered frequently through ASP (Active Server Pages) reporting.

Contents

Introduction

The error, “Server Has Not Yet Been Opened” is generally encountered when an ASP application is not correctly logging on to the data source to which the report is attempting to connect. There are few different situations that may cause this error and this document discusses how to troubleshoot them.

Troubleshooting Steps

The following is a list of troubleshooting steps to be taken in order to resolve the error “Server Has Not Yet Been Opened”.

1. Verify Database Connectivity from the Designer

When working with an ASP application, or any Crystal Web application, the most important thing to verify is whether the report can connect to the database from within the Crystal Reports designer **installed on the web server computer**.

i) Open the report in the designer **on the web server computer**.

ii) Select “Verify Database” from the “Database” menu. If the connection to the database is successful the message, “Database is up to date” will be displayed. If another message, or even no message at all, is displayed it may be an indication that there is a problem with the database connection for this report.

NOTE	If the report is connecting to the database through an ODBC connection ensure that an ODBC System DSN exists on the web server machine (Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC)). Ensure that the DSN is a SYSTEM DSN so that there are no data base access issues for individual users.

After you have verified that the database is up to date it’s a good idea to page through the entire report from the designer to ensure that each page of the report appears correctly. Any error messages that you encounter while previewing the report within the designer will also affect how the report performs through the web components.

2. Verify Connection Parameters from the Designer

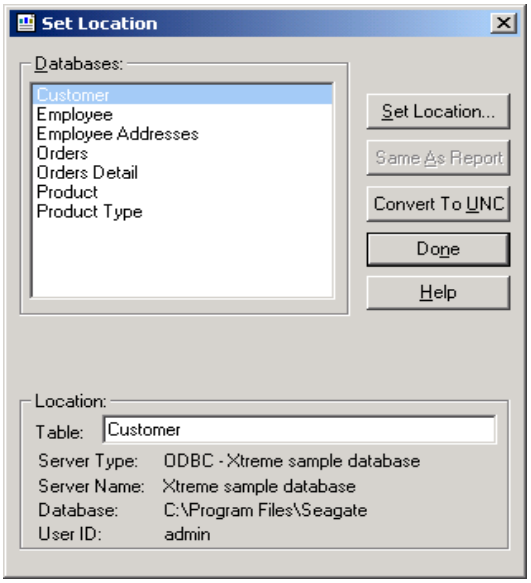
For database security reasons the report designer does not save database passwords within a report’s definition. Therefore, the UserID and Password must always be provided when an ASP application is connecting to a secured data source.

It is essential that the ASP application is attempting to pass the same data source name, database name, location, credentials, etc (some or all of these attributes may be needed depending on the specific database) as the report is expecting.

NOTE	You must ensure that the data base connection attributes are passed exactly as seen in the "Database > Set Location" box in the designer. **Case sensitivity is essential.**

You may want to write down the database credentials that the report is using from within the designer. A common cause of the "Server has not yet been opened" error message is not matching the database credentials that the report is expecting. The "Set Location" window under the Database menu item provides this information for you.

It is a good idea to keep case sensitivity when passing logon credentials over the web.



In the example above you will notice that the Set Location Window provides you with the first 3 database credentials for the report. The "ServerName" represents the name of the ODBC DSN for the report. The name of the database is located next to the "Database:" line. As well the UserID for the database is stored next to the "UserID:" line. It is a good idea to record this information, as it will be the same information that you will later need to provide in your ASP code.

Further troubleshooting information for resolving designer related connection issues can be found in the technical brief "cr_dbconn_troubleshooting.pdf", which can be downloaded at:

http://support.crystaldecisions.com/docs

3. Address known ODBC connection issues

In the original release of Crystal Reports 8.0 (build #8.0.0.371) there was a known issue with ODBC data sources and registry permissions. This issue was assigned the Track ID #36747, and is resolved in Crystal Reports build 8.0.0.441 and higher.

The issue was that existing ODBC registry permissions were modified during the Crystal Reports 8.0 installation. ODBC connections created after the installation of this version of Crystal Reports would also not have the correct user permissions associated with them but we'll talk about this in more depth below. For more information on the symptoms and file downloads available to fix this issue please consult Knowledge Base article c2007581, which can be found through searching for the article number at:

http://support.crystaldecisions.com/kbase

4. Fix existing version 8 ODBC Connections

Crystal Reports build # 8.0.0.441 and higher no longer introduce the ODBC DSN permissions issue for new ODBC data entries. However, DSNs created prior to the installation of version 8.0.0.441 will still need to be repaired. Therefore, it is important that the file, **Scr8_webregfix.exe,** be applied to the web server computer to resolve the permissions related issues with existing ODBC data sources.

This file can be downloaded from our support site:

http://support.crystaldecisions.com/updates

Filename: **Scr8_WebRegFix.exe**

Scr8_webregfix.exe contains a utility that restores the ability to create ODBC connections that give access to the "Everyone" group of Windows users. Connections created after this utility has been used will not encounter this issue. This patch restores ODBC registry permissions to the Microsoft Windows NT/2000 default registry permissions.

It is also a good idea to run the Crystal Decisions Database and Export Drivers Patch along with the utility above to ensure that the latest ODBC database drivers are installed.

This file, Dbexdrvr.zip, can also be downloaded from the Support site at:

http://support.crystaldecisions.com/updates

5. Verify the ODBC data source configuration

Open the ODBC Configuration manager (Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC)) on the webserver computer and verify that the DSN (Data Source Name) is setup **only as a System DSN**. Ensure that there is not any **File** or **User DSNs** with the same data source name. If so these must either be removed or renamed.

6. Verify Connection Information in the ASP Application

There are several different methods that you can use to pass logon credentials to your database at runtime. It is strongly recommended that you use the "SetLogonInfo" method to logon to your database rather than "LogonServer".

"LogonServer" creates a connection to the entire database and server and will leave this connection open for the entire duration. This will cause a considerable resource drain on your database server. It has also been shown to leave open database connections after the session has been closed which can be considered a security issue.

As well logonserver has been shown to be less stable and less reliable in its results.

Setlogoninfo, connects to the database at the table level and so this will definitely reduce overhead. As well, it is more reliable and consistent.

Here is the typical block of code involved with the SetLogonInfo method:

```
UserID = "Your database userID goes here"
Password="Your database password goes here"
Set crtable = session("oRpt").Database.Tables.Item(1)
Crtable.SetLogonInfo "DSN_Name", "Database_Name",
cstr(UserID), cstr(Password)
```

Note: With native database connections the first variable in the SetLogonInfo method is replaced with the ServerName of the machine the database exists on.

Also, Oracle data sources do not use database names therefore, with Oracle datasources the "Database_Name" parameter is replace with "".

For further sample code with regards to the SetLogonInfo method please download our ASP samples "Aspxmps85.exe" from our support site at:

http://support.crystaldecisions.com/updates

Testing connectivity with the SetLogonInfo method

An additional line of code you can be added to the above code block for error checking purposes. Immediately following the code block above add the following line of code:

```
Response.write crtable.testconnectivity
```

If the "**testconnectivity**" test on the table object connectivity method returns "True" then the connection to the table has been established.

If the "testconnectivity" test returns "False" there is no database connectivity and you should ensure that the SetLogonInfo line of code is correct.

7. Check for Mismatched File Versions

Mismatched files on the web server computer can also create connectivity issues. Search for P2sodbc.dll on the web server computer, make sure there is only one version of the file, and that it is version 8.x.x.xxx. Also search for the Craxdrt.dll and make sure there is only one version and it begins with version 8.x.x.xxx.

8. Native Database Connections

The Crystal Reports RDC (Report Designer Component) does not provide the ability to change the connection type at runtime. For instance it is not possible

to change a report from using a native SQL connection to a native Oracle connection at runtime.

It is also not possible to change between databases of different types with native connections. For instance it is not possible to change from a native SQL connection to a native Oracle connection at runtime.

It is however, possible to change between two of the same native data sources at runtime i.e. Native SQL database A -> Native SQL database B.

9. RDC and Windows Security Contexts

On the Microsoft MSDN library there is an excellent article entitled "Troubleshooting Components Under ASP Technology".

The Microsoft MSDN site is located here:

http://msdn.microsoft.com

One of the topics the Microsoft article discusses is ASP and Windows Security contexts. This topic has a significant impact on several users receiving the "Server Has Not Yet Been Opened" error message.

Under the "Security Issues" section of the document, component security is discussed. It is important to note that and single-threaded or free-threaded components will run under the context of the local system account. However, any ASP component that is either Apartment model threaded or multi-threaded will run under the context of the Internet Guest account IUSR_MachineName.

When reporting through ASP, using Crystal's Report Designer Component (Craxdrt.dll) it is important to know that the threading model is apartment model threaded. This means that it will be operating under the IUSR context. If that account is limited in it's access to connectivity DLLs or DSNs then we may see connectivity issues. It is important to ensure that the Internet Guest account has sufficient permissions to achieve connectivity.

Additionally, if any part of the connectivity process is reduced to a single threaded model (i.e. a non thread safe ODBC driver or native connectivity DLL) then the permissions context will be reduced to that of the System account so it is also important to ensure that account has appropriate permissions on the system.

For assistance with rights and permissions, we have white papers that outline the minimum required NTFS permissions for web reporting (search for "c2008225" on our Support Site at http://support.crystaldecisions.com/kbase). The application of this document can often help to resolve connectivity issues.

This impacts Crystal ASP applications as depending on the threading model you are bound to either the local system account or the IUSR_MachineName account will have to have valid permissions to the database server in order to authenticate correctly.

A workaround is to create an MTS/COM+ package to run the RDC under the context of another account, such as an administrative account.

For instructions on how to place Craxdrt.dll (RDC) into an MTS package please see support article c2006450 for Windows NT instructions and c2009021 for Windows 2000 instructions.

The online knowledge base support site is located at the URL below:

http://support.crystaldecisions.com/library/kbase.asp

10. "Server has not yet been opened" and Windows NT Authentication

A report connected to an ODBC database using a DSN that is authenticating through NT Authentication can often cause the error "Server has not yet been opened".

Please note that Microsoft is not able to authenticate using NT Authentication to any database server that does not reside on the same machine as the web server. This concept is referred to as "Delegation". For more information on "Delegation" and IIS/Internet security concepts please see Microsoft's MSDN support site. You can find the site at the URL below:

http://msdn.microsoft.com

The only known workaround is to use SQL Server Authentication when the database server resides on another machine.

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/

Answers By Email Support:

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

Phone Support:

Tel: (604) 669-8379