

Crystal Enterprise 8

URL Commands for the Web Component Server

Overview

This document discusses the various functions that can be performed on a web report using the URL commands available through the Crystal Web Component Server (WCS).

Contents

INTRODUCTION	3
SPECIFYING THE ID OF THE REPORT	4
<i>ID</i> command.....	4
SPECIFYING THE APS LOGON TOKEN	5
<i>APSTOKEN</i> command.....	5
SPECIFYING APS LOGON INFORMATION	6
<i>APSUSER, APSPASSWORD, APSAUTHTYPE</i> commands	6
SPECIFYING THE REPORT VIEWER	7
<i>INIT</i> command	7
CONNECTING TO THE PAGE SERVER	8
<i>CONNECT</i> command.....	8
LOGGING ONTO YOUR DATABASE	9
<i>USER and PASSWORD</i> Command.....	9
Method 1 – <i>USER# and PASSWORD#</i> command to pass logon credentials to the main report and subreport	9
Method 2 - <i>USER and PASSWORD</i> command to pass logon credentials, server name and database name.....	11
PASSING PARAMETER VALUES TO THE REPORT	12
<i>PROMPTEX</i>	12
Method 1 – Using <i>PROMPTEX</i>	12
Method 2 – Using the <i>PROMPTEX</i> command to pass multiple values to a parameter	13
Method 3 – Using <i>PROMPTEX#</i>	15
Method 4 – Using <i>PROMPTEX</i> to Pass Date Values.....	15
<i>PROMPT#</i>	17
Method 1 – Using <i>PROMPT#</i> to Pass Single Parameter Values.....	17
Method 2 – Using <i>PROMPT#</i> to Pass Date Values	18

PARAMETER VALUES ON REFRESH 18
PromptOnRefresh command..... 18

GROUP AND RECORD SELECTION FORMULAS..... 19
SF command 19
GF command..... 19

EXPORTING REPORTS 20
CMD and EXPORT_FMT commands..... 20
Refreshing the data while exporting 21

FINDING MORE INFORMATION 21

Introduction

In Crystal Enterprise you can append several Report Viewer commands to a Crystal Web Request (CWR). These commands allow you to control the way reports are generated and displayed. For example, instead of the user being prompted for information, you can automatically assign values for database authentication, parameters, selection formulas as well as other features.

Report Viewer commands can be passed in any order and in any combination. All commands are optional—except for the ones that are needed to launch the Report Viewer. If you do not specify any optional commands, the default viewer displays the report and prompts the user for any required information.

Keep in mind that there are a number of factors that determine whether or not the user will be prompted for information. The user will be prompted under the following circumstances:

- The report requires the user to enter parameter values or authentication information.
- The report does not contain saved data; therefore it needs to access a database.
- The user has refreshed the report; therefore it needs to access a database.
- The values for the prompts have not already been set or the prompts have been enabled through the SDK or Crystal Management Console.

Below is a table, which lists all of the available Report Viewer commands:

Command	Description
ID	Specifies the ID of the current report.
APSTOKEN	Specifies the logon token for the current APS session.
APSUSER, APSPASSWORD, APSAUTHTYPE	Specifies the APS user name, password, and authentication type.
INIT	Specifies the Report Viewer.
CONNECT	Re-establishes a connection to the Page Server.
PASSWORD	Specifies the passwords for logging on to SQL, ODBC, or password-protected databases used by the report.
USER	Specifies the user IDs for logging on to SQL or ODBC databases used by the report.
PROMPTEX#	Specifies values for parameter fields in a report.
PROMPT#	Specifies values for parameter fields in a report. This method of specifying parameter values is used in older versions of Crystal Reports (for example, Crystal Reports 7).

PromptOnRefresh#	Specifies whether report should prompt for parameter field values when refreshed.
SF (Selection Formula)	Specifies a selection formula.
GF (Group Selection Formula)	Specifies a group selection formula.
CMD# and EXPORT_FMT	Specifies that the report should be exported to the indicated format.

Specifying the ID of the Report

ID command

The ID command specifies the report ID of the current report.

Syntax
<code>id=<value></code>
Example
<p>The example below illustrates how CSP scripting can be used to create a link which generates a value for the report ID:</p> <pre>" View Report"</pre> <p>The resulting URL for this link is below:</p> <pre>http://myServer/myVirtualDirectory/viewrpt.cwr?id=15 2&apstoken=myAPS:A1B2</pre> <p>The ID of this report is "152", a number that allows the APS to uniquely identify each report. It corresponds with the SI_ID Property.</p>

Specifying the APS Logon Token

APSTOKEN command

CAUTION

Logon tokens are not encrypted. They consist of a set of numbers and letters that are assigned internally and then recognized by the APS.

The APSTOKEN command is used to log on to the Crystal APS. The APSTOKEN specifies the logon token for the current APS session.

Using a logon token ensures that your company's licensing information is accurate. Each time a user logs on to the Crystal Enterprise system, the count for the number of licenses currently in use increases by one. If a logon token is not passed as a parameter, each time the user logs on to the APS (for example, the user may want to log on to a different Crystal Enterprise client component), the count will increment. As a result, an administrator or auditor may overestimate the number of licenses in use.

If the APS does not authenticate a user with the APSTOKEN command, you can use the APSUSER, APSPASSWORD, APSAUTHTYPE commands to pass the necessary values to authenticate the user against the Crystal APS.

Syntax

```
apstoken=<token>
```

Example

The example below uses CSP scripting to display a report when the user clicks the View Report link.

```
"<a target='_blank' href='viewrpt.cwr?id=" &
"CurrentReportID & " & _
    apstoken=" & LogonToken & "'> View Report</a>"
```

The resulting URL for this link is below:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=15
2&apstoken=myAPS:A1B2
```

Specifying APS Logon Information

APSUSER, APSPASSWORD, APSAUTHTYPE commands

CAUTION

For security reasons, specifying the user name and password over the URL is not recommended.

The APSUSER, APSPASSWORD, APSAUTHTYPE commands are used to log on to the Crystal APS. They specify the APS user name, password, and authentication type.

You may need to use these commands under special circumstances, such as when a user receives a report through email and needs to log on to the APS in order to view it.

In most cases, however, you will want to use the APSTOKEN command to log on to the APS. Refer to the section “Specifying the APS Logon Token” for further instructions on how to use the APSTOKEN command.

Syntax

```
apsuser=<user name>  
apspassword=<password>  
apsauthtype=<authentication type>
```

Example

Below is an example that illustrates how to use the commands within a URL. The following values are specified within the URL:

- “Jlee” is specified for APSUSER
- “secret” is specified for APSPASSWORD
- “secEnterprise” is specified for APSAUTHTYPE

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=15  
2  
&apsuser=JLee&apspassword=secret&apsauthtype=secEnte  
rprise
```

Specifying the Report Viewer

INIT command

The INIT command specifies the report viewer that will be used to view the report. If the INIT command is not specified, the Crystal Web Component Server detects the type of browser requesting a report and provides a default viewer that is most appropriate for that browser.

The default viewer for Microsoft Internet Explorer is the ActiveX viewer. The default viewer for Netscape Navigator is the Java viewer.

Syntax
<code>init=<viewer></code>
Example
<p>The example below specifies that the java viewer will be used to view the report:</p> <pre>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&init=java</pre>

Below are the possible values for <viewer>:

Value	Viewer type
actx	ActiveX
nav_plugin	Netscape Plug-in
java	Java using browser JVM
java_plugin	Java using Java Plug-in
html_page	Standard HTML
html_frame	DHTML with Frames

Connecting to the Page Server

CONNECT command

The CONNECT command re-establishes a connection to the Crystal Page Server. The CONNECT command must be appended to the INIT command. By re-establishing a connection to the Page Server, the CONNECT command allows the user to reset the report's parameters and logon information, and re-process the report if necessary—without having to begin a new browser session.

If you use report viewer A to display a report, and then you specify report viewer B to view the same report in the same browser session, you will not be prompted for parameter values or database logons, and a new report job will not be opened. However, if you specify “:connect” along with the request for report viewer B, the connection to the Crystal Page Server will be re-established, which means that if necessary, the user will be prompted for parameter values and logon information, and the report will be run again.

Syntax

```
init=<viewer>:connect
```

Example

The example below specifies that the report will re-establish its connection to the Crystal Page Server once the URL has been processed:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&init=java:connect
```

NOTE

Re-establishing a connection to the Crystal Page Server using the CONNECT command is not the same as refreshing a report against a database. While the connection to the Crystal Page Server enables the user to reset parameter values and view a different set of information, if the report contains saved data, it will not need to access the database for this information. For more details on refreshing a report, download the [Crystal Enterprise Web Developer's Guide](#) for CE 8 and search for “Refreshing web report data”.

Logging onto your Database

USER and PASSWORD Command

The USER and PASSWORD command specifies the user name and password for logging on to SQL or ODBC databases used by the report. The USER command is always accompanied by the PASSWORD command. The USER or PASSWORD commands alone would render the URL useless. Below are two methods of using the USER and PASSWORD command:

Method 1 – USER# and PASSWORD# command to pass logon credentials to the main report and subreport

The USER# and PASSWORD# command allows you to pass logon credentials for the database that the report is using. This method of specifying a user name and password was used in older versions of Crystal Reports (i.e. Crystal Reports 7). This can be done for both the main report and subreport.

If an existing report is inserted as the subreport, then the subreport name includes the file extension (for example, user0@subreportname.rpt). However, if the subreport was created inside the main report (with Insert Subreport and using the Report Expert to create the new report) then the name of the subreport usually does not contain a file extension (for example, user0@subreportname) unless one is added in the "Report Name" text box of the Insert Subreport dialog box.

User names and passwords can be specified in the URL in any order; for example, user1 can appear before user0. However, index numbers must match the order that the password-protected databases appear in the report. Additionally, subreport user names and passwords must appear in the same order that the subreports appear in the report.

Reports are not shared if there are logon requirements.

Syntax

Main Report

```
user#=<user name>&password#=<password>
```

Subreport

```
user#@subreportname=<user  
name>&password#@subreportname=<password>
```

Example**Main Report**

The example below passes user name "msmith" and the password "1234" to the report:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&user0=msmith&password0=1234
```

Subreport

The example below illustrates how to pass the user name "msmith" and password "1234" to the subreport called "Crosstab":

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&user0@Crosstab=msmith&password0@Crosstab=1234
```

If the report accesses more than one password-protected database, multiple user names and passwords can be passed by incrementing the USER and PASSWORD index number, as shown below:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&user0=msmith&password0=1234&user1=bsmith&password1=1234
```

NOTE

Passwords can be specified in the URL in any order; for example, password1 can appear before password0. However, the index numbers must match the order that the password-protected databases appear in the report.

Method 2 - USER and PASSWORD command to pass logon credentials, server name and database name

The USER and PASSWORD command can be used to pass logon credentials as well as the server name and database name. This can be done for both the main report and subreport.

<p>Syntax</p>
<p>Main Report</p> <p>User-<servername>.<databasename>& Password-<servername>.<databasename></p> <p>Subreport</p> <p>User-<servername>.<databasename>@<subreportname>& Password-<servername>.<databasename>@<subreportname></p>
<p>Example</p>
<p>Main Report</p> <p>The example below illustrates how to pass the following values to the report:</p> <p>server name "systemdsn" database name "xtreme"</p> <p>NOTE: For Oracle databases substitute the schema name for the database name</p> <p>user name "vantech" password "1234"</p> <p>http://myServer/myVirtualDirectory/viewrpt.cwr?user-systemdsn.xtreme=vantech&password-systemdsn.xtreme=1234</p> <p>Subreport</p> <p>The example below illustrates how to pass the following values to the subreport:</p> <p>server name "systemdsn" database name "pubs" user name "vantech" password "1234" subreport "sr"</p> <p>http://myServer/myVirtualDirectory/viewrpt.cwr?user-systemdsn.pubs@sr=vantech&password-systemdsn.pubs@sr=vantech</p>

Passing Parameter Values

The PROMPTEX and PROMPT are two URL commands that allow you to pass a parameter value to a report. You can use either the PROMPTEX or the PROMPT commands. This section will detail how to use each method.

PROMPTEX

The PROMPTEX command specifies values for parameter fields in a report.

Method 1 – Using PROMPTEX

The PROMPTEX command allows you to specify parameter by name as well as the value for the parameter.

Syntax

Main Report

```
promptex-<promptname>=<value>
```

Subreport

```
promptex-<promptname>@<subrpt>=<value>
```

<promptname> and **<subrpt>** are non-empty strings that represent names of a parameter field prompt and a subreport that are defined in the report.

<value> is a single or multivalued string.

Example

Main Report

In the example below “hello” is being passed as a value for the parameter called “sample”:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&promptex-sample="hello"
```

Subreport

In the example below “hello” is being passed as a value for the parameter called “sample” for the subreport called “mysubrpt”:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&promptex-sample@mysubrpt="hello"
```

NOTE	<ul style="list-style-type: none">▪ If an existing report is inserted as the subreport, then the subreport name includes the file extension (.rpt).▪ A backslash \ acts as an escape, so it is substituted by the character that follows it. Quotation marks and backslashes need to be "escaped" as they are reserved URL characters.
-------------	---

Method 2 – Using the PROMPTEX command to pass multiple values to a parameter

The PROMPTEX command can be used to pass simple multiple values for a single parameter. Or it can be used to pass a range of multiple values for a single parameter. Both methods are outlined below:

Simple Multiple Values

The PROMPTEX command allows you to specify multiple values to a parameter.

Syntax
<code>promptex-sample="<value A>","<value B>","<value C>"</code>
Example
<p>The example below specify "Apples, Oranges and Grapes" as values for the parameter called "fruits":</p> <pre>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&promptex-fruits="Apples","Oranges","Grapes"</pre>

Range of Multiple Values

The PROMPTEX command allows you to specify multiple values to a parameter.

Syntax
<code>promptex-sample=["<value A>" - "<value B>"]</code>
Example
<p>A square bracket indicates that the interval is closed at that end, and that the specified number is included in the range; a round bracket indicates that the interval is open at that end, and that the specified number is not included in the range.</p> <p>For example:</p> <p><code>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&promptex-sample=("5" - "11")</code></p> <p>Specifies a range of all values between 5 and 11, not including 5 and 11 themselves.</p> <p><code>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&promptex-sample=["5" - "11"]</code></p> <p>Specifies a range of all values between 5 and 11, including 5 but not 11.</p> <p><code>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&promptex-sample=(- "11")</code></p> <p>Specifies a range of all values up to, but not including, 11.</p>

The following lists the types of bounded and unbounded intervals you can use.

Bounded interval	Unbounded intervals
<code>["<value>" - "<value>"]</code>	<code>("<value>" -)</code>
<code>("<value>" - "<value>"]</code>	<code>["<value>" -)</code>
<code>["<value>" - "<value>")</code>	<code>(- "<value>")</code>
<code>("<value>" - "<value>")</code>	<code>(- "<value>"]</code>

Method 3 – Using PROMPTEX#

The PROMPTEX# command is an enhanced version of the older Prompt# command. In the enhanced notation, quotation marks are used around parameter values to indicate string values. All parameter values are passed to the report as strings, and intended numeric values are translated from strings to numbers by the report.

Syntax

```
promptex#=<value>
```

Example

The example below illustrates the PROMPTEX# method passing "CA" as a value for the first parameter in the report:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&promptex0="CA"
```

If the report contains more than one parameter field, multiple values can be passed to parameters by incrementing the PROMPT index value. For example:

```
promptex0="CA"&promptex1="1000"
```

Prompts can be specified in the URL in any order; for example, `promptex1` can appear before `promptex0`. However, index numbers must match the order that the prompts appear in the report.

Method 4 – Using PROMPTEX to Pass Date Values

In order to pass Date or DateTime parameter values over the URL, use the Single Value or Date Range methods as specified below.

Single Value Date or DateTime Parameters

For single value Date or DateTime parameters the `promptex-<promptname>` command requires double quotes.

Syntax

```
promptex-<promptname>="Date ( YYYY , MM , DD ) "
```

Example

To pass a Date value of February, 02, 2002 for the "birthdate" parameter the URL command you will use will be:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=15
2&apstoken=A1B2&promptex-
birthdate="Date(2002,02,02)"
```

Date Ranges

For passing date ranges you will need to use square brackets [].

Syntax

```
promptex-<promptname>=[ "Date ( YYYY , MM , DD ) " -
"Date ( YYYY , MM , DD ) " ]
```

Example

The example below illustrate that "DateRangeParameter" is the parameter name, the square brackets surrounding the values indicate that the specified date is included in the range:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=15
2&apstoken=A1B2&promptex-
DateRangeParameter=[ "date(1996,02,18)" -
"Date(1996,09,10)" ]
```

The type of brackets you use surrounding the date value can specify whether the value should be included or excluded from the date range. The following will list the importance of bracket type:

- Square brackets [] surrounding the values indicate the specified date is included in the range
- Round brackets () surrounding the values indicate the specified date is excluded in the range.

NOTE

Reports that have the PROMPTEX# parameter applied do not have their pages shared. Caching will be by user.

PROMPT#

Specifies values for parameter fields in a report. This method of specifying parameter values is used in older versions of Crystal Reports (for example, Crystal Reports 7). While it is not recommended, it can still be used.

Method 1 – Using PROMPT# to Pass Single Parameter Values

PROMPT# can be used to specify parameter values.

Syntax
<pre>prompt#=<value></pre> <p><value> is a string. This arbitrary (and potentially empty) string is the new value of the prompt. Values are assigned to parameters in the order in which they exist in the report.</p>
Example
<p>In the example below "CA" is the value being passed to the first parameter:</p> <pre>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&prompt0=CA</pre> <p>If the report contains more than one parameter field, multiple values can be passed to parameters by incrementing the PROMPT index value. For example:</p> <pre>prompt0=CA&prompt1=1000</pre> <p>Prompts can be specified in the URL in any order; for example, <code>prompt1</code> can appear before <code>prompt0</code>. However, index numbers must match the order that the prompts appear in the report.</p>

NOTE	The Prompt# command is an older version of the Promptex# command. Do not use quotation marks around parameter values to indicate string values. All parameter values are passed to the report as strings, and intended numeric values are translated from strings to numbers by the report.
-------------	---

NOTE	Reports that have the PROMPT# command applied do not have their pages shared. Caching will be by user.
-------------	--

Method 2 – Using PROMPT# to Pass Date Values

For single value Date or DateTime parameters the PROMPT# command does not require double quotes.

Syntax
<code>prompt#=Date(YYYY,MM,DD)</code>
Example
To pass a Date value of February, 02, 2002 for the second parameter within a report, the URL command you will use will be: <code>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&prompt2=Date(2002,02,02)</code>

Parameter Values on Refresh

PromptOnRefresh command

The PromptOnRefresh command specifies whether the report should prompt for a parameter field values when refreshed.

Syntax
<code>promptOnRefresh=<value></code>
Example
PromptOnRefresh will only accept 0 or 1 as a value. 0 for false and 1 for true. <code>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&promptOnRefresh=1</code>

Group and Record Selection Formulas

SF command

The SF command specifies a selection formula for the report.

Any selection formula passed via the URL using the SF command will be appended to any selection formulas already contained in the report; that is, the generated report will be based on existing selection formulas AND the newly specified SF command. For example, if the report already contains a selection formula that selects the records for film studios in the state of California, and then the SF command is used to append a formula that selects the records for "Universal," information on that particular studio will be displayed. If the SF command had specified a value such as "Disney," and this studio was located only in the state of Florida, then the requested report would contain no data.

Syntax
Sf=<formula>
Example
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&sf={studio.Studio}&%3d'Universal'

NOTE	The new selection formula is not saved with the original report file. It is only valid for the currently requested job.
-------------	---

GF command

The GF command specifies a group selection formula for the report. This command is similar to the selection formula SF command.

Syntax
gf=<formula>
Example
The example below illustrates how to pass a group formula which selects all groups in which the sum of all customer sales in each region is greater than 10,000:
<pre>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&gf=Sum({customer.Sales},{customer.Region})>10000</pre>

NOTE

Reports that have exactly the same SF and GF commands applied and do not require logon information will share pages.

Exporting Reports

CMD and EXPORT_FMT commands

The CMD and EXPORT_FMT commands specify that the report should be exported to the indicated format.

Syntax

```
cmd=EXPORT&EXPORT_FMT=<EXPORT_FMT representation>
```

Example

If a user would like the report downloaded to their browser in Rich Text Format (RTF), the URL would be:

```
http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&cmd=EXPORT&EXPORT_FMT=U2FRTF%3A0
```

Below is a listing of possible values for the <EXPORT_FMT representation>:

Export Format	Export_FMT Representation
Adobe PDF	U2FPDF:0
HTML 3.2	U2FHTML:2
HTML 4.0	U2FHTML:3
Crystal Reports (RPT)	U2FCR:0
Excel 5.0 (XLS)	U2FXLS:3
Excel 5.0 (XLS) Extended	U2FXLS:4
Rich Text Format (RTF)	U2FRTF:0
Word Document (DOC)	U2FWORDW:0
Extensible Markup Language (XML)	U2FXML:0

Refreshing the data while exporting

The connect=1 command at the end of the URL command ensures that a live database hit is made when exporting to a desired format.

Syntax
connect=1
Example
<pre>http://myServer/myVirtualDirectory/viewrpt.cwr?id=152&apstoken=A1B2&cmd=EXPORT&EXPORT_FMT=U2FRTF%3A0&connect=1</pre>

Finding More Information

For more information and resources, refer to the product documentation and visit the support area of the web site at: www.businessobjects.com

► www.businessobjects.com

Business Objects owns the following U.S. patents, which may cover products that are offered and sold by Business Objects: 5,555,403, 6,247,008 B1, 6,578,027 B2, 6,490,593 and 6,289,352. Business Objects, the Business Objects logo, Crystal Reports, and Crystal Enterprise are trademarks or registered trademarks of Business Objects SA or its affiliated companies in the United States and other countries. All other names mentioned herein may be trademarks of their respective owners. Product specifications and program conditions are subject to change without notice. Copyright © 2004 Business Objects. All rights reserved.