← Click on left for Bookmarks

DataLink Viewer

View, Print, and Export Crystal Reports

www.MilletSoftware.com

Version 6.5002
August 2016

By

Millet Software
5275 Rome Ct.
Erie, PA 16509-3951
ido@MilletSoftware.com
(814) 825-6009

Disclaimer: This software is provided "as-is" by Millet Software without assuming any responsibility for harm to computer systems, software, or data with which these files are used.

INTRODUCTION **6**

MAIN BENEFITS: **6**

INSTALL / REMOVE **8**

CHECK FOR UPDATES **9**

TYPICAL USE SCENARIOS **10**

USER INTERFACE 10

SELECT A REPORT TO VIEW 12

SELECTIVE PARAMETER REFRESH 13

REMEMBERING PARAMETER VALUES 14

SAVE AND REUSE NAMED PARAMETER SETS 15

DYNAMIC & CASCADING PARAMETERS **17**

RESPOND TO SINGLE-VALUE LIVE PROMPTS 17

LINK BACK IN A HIERARCHY OF LIVE PROMPTS 18

RESPOND TO MULTI-VALUE LIVE PROMPTS 19

MANAGE DISPLAY & DATA VALUES FOR LIVE PARAMETERS 19

VIEW THE REPORT 21

REMEMBER PARAMETER VALUES FROM A PREVIOUS SESSION 21

LAUNCH REPORTS FROM FILE EXPLORER **22**

LAUNCH REPORTS FROM COMMAND LINES **23**

EXAMPLES 23

Call DataLink Viewer from Another Application 24

REFERRING TO SAVED ENCRYPTED PASSWORDS 25

COMMAND LINE ARGUMENTS FOR PARAMETER VALUES 26

Range and Multi-Value Parameters 26

Null Values 26

ARGUMENT FOR SETTING FORMULA EXPRESSIONS 27

COMMAND LINE ARGUMENTS FOR TRIGGERING/SCHEDULING EXPORTING 28

Interactive Use 28

Fully-Automated Exporting 28

Scheduling Exporting 28

COMMAND LINE ARGUMENTS FOR TRIGGERING PRINTING 29

SPECIFYING NUMBER OF COPIES 29

SETTING CUSTOM TEXT FOR EACH PRINT COPY 29

SCHEDULING PRINTING 30

Scheduling Printing for Multiple Reports 30

LAUNCH A REPORT WHILE VIEWING ANOTHER REPORT 31

CREATE A USER INTERFACE FOR SELECTING AND LAUNCHING REPORTS 32

EMBED INPUT FROM THE USER IN THE COMMAND LINE CALL FOR ANOTHER REPORT 33

LAUNCH ANOTHER APPLICATION AND PASS PARAMETERS TO IT 34

LAUNCHING REPORTS IN A NEW WINDOW 35

LAUNCH A REPORT IN A NEW WINDOW 35
New Window On File Launch and/or On Double Click 35
Right-Click the Grid and Select Preview Report (new Window) 36

REMOVE THE PREVIEW TAB FROM THE INITIAL WINDOW 36

CONTROL DATA ACCESS ACCORDING TO PC LOGIN 37

LOAD INI VALUES INTO PARAMETERS 38

SECURING REPORTS AGAINST UNAUTHORIZED USE 39

LOADING MACHINE INFORMATION INTO PARAMETERS 39
DLV_Machine_Name 39
DLV_Registered_Company 39
DLV_HD_Serial_N 39

DATABASE CHOICE FUNCTIONALITY 40

SELECT ALTERNATIVE ODBC DATA SOURCES FOR THE SAME REPORT 40
Forcing the Login Dialog to Display to Allow Choice of ODBC DSN 41
Restricting the List of ODBC DSN Choices 41

USING COMMAND LINE ARGUMENTS TO SPECIFY THE ODBC DSN 42

OVERRIDING THE DATABASE SPECIFIED IN THE REPORT OR ODBC DSN 43

STRIPPING TABLE QUALIFIERS WHEN CONNECTING TO A DIFFERENT DATABASE 44

OVERRIDING THE SERVER IN NATIVE ORACLE CONNECTION 44

SELECTING AN ALTERNATIVE SQL SERVER – OLE DB DATA SOURCE 44

FORCED LOGIN 45

SELECTING FOLDER LOCATION FOR FOXPRO DBF FILES (MASTERBUILDER) 46

CHANGING FOLDER LOCATION FOR ACCESS/EXCEL/PARADOX FILES 47
Using Command Line Argument to Avoid Path Prompt 47

INTEGRATED AUTHENTICATION 48

INTEGRATED AUTHENTICATION ("REMEMBER ME") 48

SHARED MACHINE AUTHENTICATION 49

SHARED SECRET PASSWORD WITH WINDOWS USER IDS 50

MONITORING DATALINK VIEWER USE 51

RECORD PROCESSING TO AN ODBC DATABASE 51
Typical Use: 51
MS Access Data Structure 52
SQL Server Data Structure 52
Oracle Data Structure 54
How to Start Logging? 55

RECORD REPORT USE IN A TEXT LOG FILE 56

PROTECT REPORT DESIGNS WITH RPZ FILES 57
Make_Rpz Command Line Argument 58

CREATE RPZ FILES WITH EXPIRATION AND LICENSE KEYS 59
Make_Rpz2 Command Line Argument 60
ADD YOUR COMPANY INFO TO THE ABOUT DIALOG 61
AUTO-REFRESH REPORTS EVERY N SECONDS 62
AUTO_REFRESH COMMAND LINE ARGUMENT 63
VIEWMODE COMMAND LINE ARGUMENT (REMOVE TOOLBAR/STATUS BAR) 63
CYCLING THROUGH SEVERAL AUTO-REFRESHED REPORTS 64
CLICK TO CHANGE A PARAMETER VALUE 65
Toggling Between 2 Parameter Values 65
Special Date Parameter Dialog 65
Input Box Dialog 65
ListBox Dialog 66
Changing a Parameter by Clicking on a Related Formula 66
CLICK TO SET FORMULA VALUE 67
CLICK COLUMN HEADERS TO RE-SORT THE REPORT 67
IN-PLACE DRILL-DOWN 68
CONVERTING SECTION DOUBLE-CLICK TO IN-PLACE DRILL-DOWN 70
DISABLING REPORT PREVIEW BUTTONS 71
DISABLING DATALINK VIEWER BUTTONS 72
CITRIX AND FILE LOCATION FUNCTIONALITY 73
Explicit Assignment of Default Printer 74
Setting Integrated Authentication for Multiple Machines (Citrix Farm) 75
Setting Encrypted Password Entries 76
FILE LOCATION REDIRECT LOGIC FOR DATALINK_VIEWER.INI 77
Checking and Navigating to Key File Locations 78
ENFORCING SETTINGS IN A MASTER DATALINK_VIEWER.INI FILE 79
UPDATING DATALINK_VIEWER.INI VIA A DELTA FILE 80
DYNAMIC AND CASCADING PARAMETERS (DEVELOPER NOTES) 81
NAMING PARAMETERS 81
DESIGNING LIVE PROMPT REPORTS 81
IMPLEMENTING A “SELECT ALL” OPTION 82
SHARING VALUES ACROSS CASCADING PARAMETERS 83
CONTROLLING THE DEFAULT VALUE IN DYNAMIC PARAMETERS 84
REQUIRING A VALUE FOR A DYNAMIC PARAMETER 85
FORCING USERS TO SELECT FROM THE LIST OF VALUES (NO DIRECT EDIT) 85
UPDATE HISTORY 86
VERSION 6.5002 RELEASED AUGUST 29, 2016 86
VERSION 6.5001 RELEASED NOVEMBER 25, 2015 86
VERSION 6.4001.0: RELEASED NOVEMBER 27, 2014 86

VERSION 6.3001.0: RELEASED SEPTEMBER 21, 2013 87
VERSION 6.2001.0: RELEASED MAY 30, 2011 89
VERSION 6.0001.0: RELEASED JANUARY 19, 2010 92
VERSION 5.9001.0: RELEASED MAY 08, 2009 94
VERSION 5.8001.0: RELEASED 7/25/2008 96
VERSION 5.7001.0: RELEASED 2/26/2008 98
VERSION 5.6001.0: RELEASED 9/23/2007 99
VERSION 5.5001.0: RELEASED 1/12/2007 101
KNOWN ISSUES AND LIMITATION 130

Introduction

While the creation and design changes of Crystal reports (.rpt files) require the full **Crystal Reports** software, you can let other PCs view, print, and export these reports by installing **DataLink Viewer**.

DataLink Viewer provides several useful features such as **command line API** (allowing you to **schedule printing** and **trigger viewing** of reports from your application, task scheduler, batch files, or desktop shortcuts), an intuitive Grid for **organizing and selecting previously opened reports**, reduced login frustrations via **integrated authentication**, choice of **alternative data sources**, **selective parameter refresh**, **dynamic and cascading parameters** even for versions prior to XI, **auto-refresh**, **user-based row-level security**, and more...

Main Benefits:

1. **View, Print, and Export** Crystal reports

2. **Schedule Report Printouts & Exports**

3. **Intuitive Grid Interface** to classify, organizing, and launch Reports.

4. **Dynamic & Cascading Parameters** (even for pre-XI reports):

   - **Select Live Parameter Values** from linked Crystal reports that act as dynamic pick list data sources or default to using Crystal's static parameters.

   - **Restrict Live Parameter Values** based on choices in a prior live parameter.

   - **Remember Values last used for each Live Prompt** allowing the user to accept or replace those values.

5. **Command Line Interface:** for launching reports from any other program and even from within other Crystal reports.

6. **Protect & Hide Report Designs** by converting your rpt files into rpz files. Your users can run the resulting rpz files in DataLink Viewer, but cannot view or modify them in Crystal. To completely protect report designs, DataLink Viewer blocks exporting of rpz files to rpt or report definition files.

7. **Filter Data based on User Login**.

8. **Export & Print Reports Directly** (without previewing)

9. **Auto-Refresh Reports**

10. **React to User Actions in Useful Ways (requires special formulas):**

    - **Click a Parameter to Change Its Value**
    - **In-Place Drill Down** (www.milletsoftware.com/Download/DLV_InPlace_DrillDown.wmv)
    - **Click to Group/Sort** (http://www.milletsoftware.com/Download/DLV_Sort_By_Click.wmv)
    - **Click to set a formula value**
    - **Click to prompt for and pass a value to another process**

11. **Compatibility with document management systems** such as Documentum.

12. **Selective Parameter Refresh:** When refreshing a report, **users can select which parameters they wish to change**. This avoids tedious re-entering of values for the other parameters. The parameter refresh choices are stored for each report and can be easily reused or changed at a later session.

13. **Integrated Authentication:** use Windows User Login and Machine ID to remove the need to repeatedly authenticate to data sources.

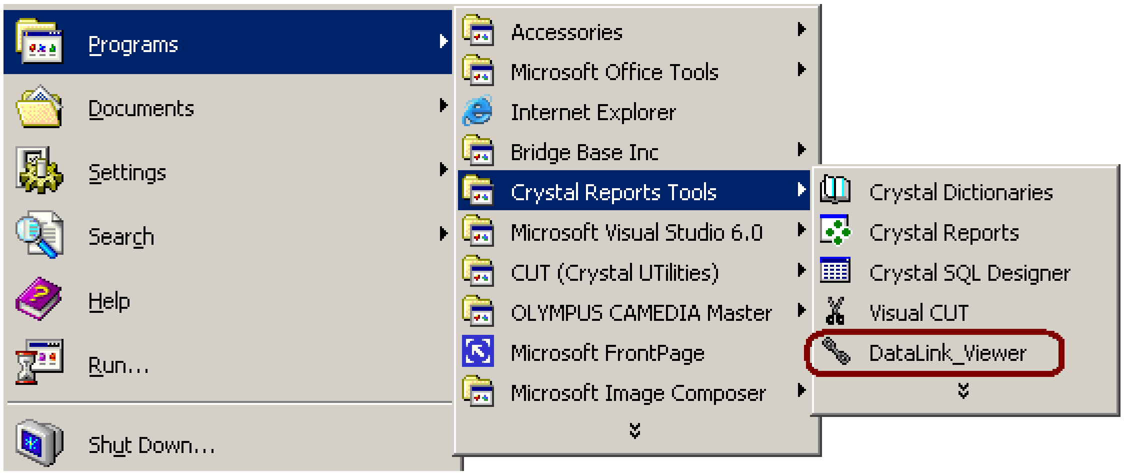
14. **Select Different Data Sources** for the same Report.

Install / Remove

The zip file you received should contain a **DataLink_Viewer.msi** (*Microsoft Installer*) file containing all the files to be installed, sample reports, and this document. Double-Clicking the **.msi** file will start the installation process, which takes care of registering the dll's and, unless you specify another location, defaults to installing the application in the ***C:\Program Files\DataLink_Viewer*** directory.

Double-Clicking the **DataLink_Viewer.msi** file again after the installation, allows you to Remove the software.

DataLink_Viewer is automatically installed under the "Crystal Reports Tools" Programs Start menu. If Crystal is not installed on the PC, the "Crystal Reports Tools" menu item will be added and ***DataLink_Viewer*** will be the only entry under it.



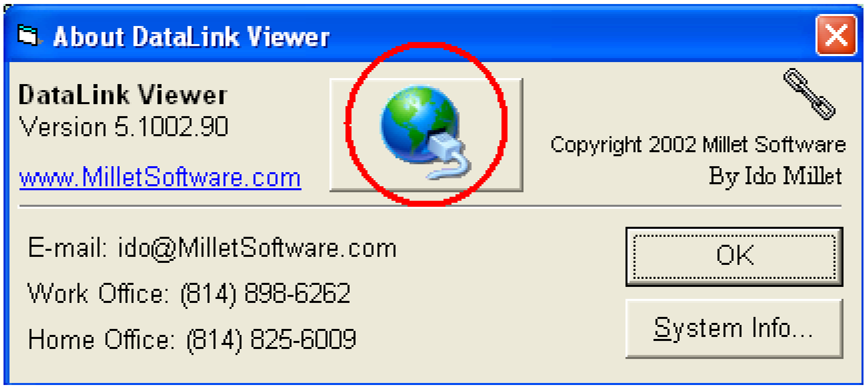
Installation on Vista:
If you run Vista, you can avoid unexpected behaviors
(see: http://www.west-wind.com/WebLog/posts/5584.aspx) by:

1. **Giving users full permissions on the DataLink Viewer application folder** (e.g., c:\Program Files\DataLink Viewer 11)
2. Right-clicking the **exe file,** select **Properties**, **Compatibility** tab, and turn on the '**Run this program as an administrator**' option

Avoid Installation on a Crystal Enterprise Machine:
Due to the risk of rare runtime component conflicts, you should avoid installing the software on a machine that also runs Crystal Enterprise.

Check for Updates

When clicking on the version information button you gain access to a button for checking for online updates:

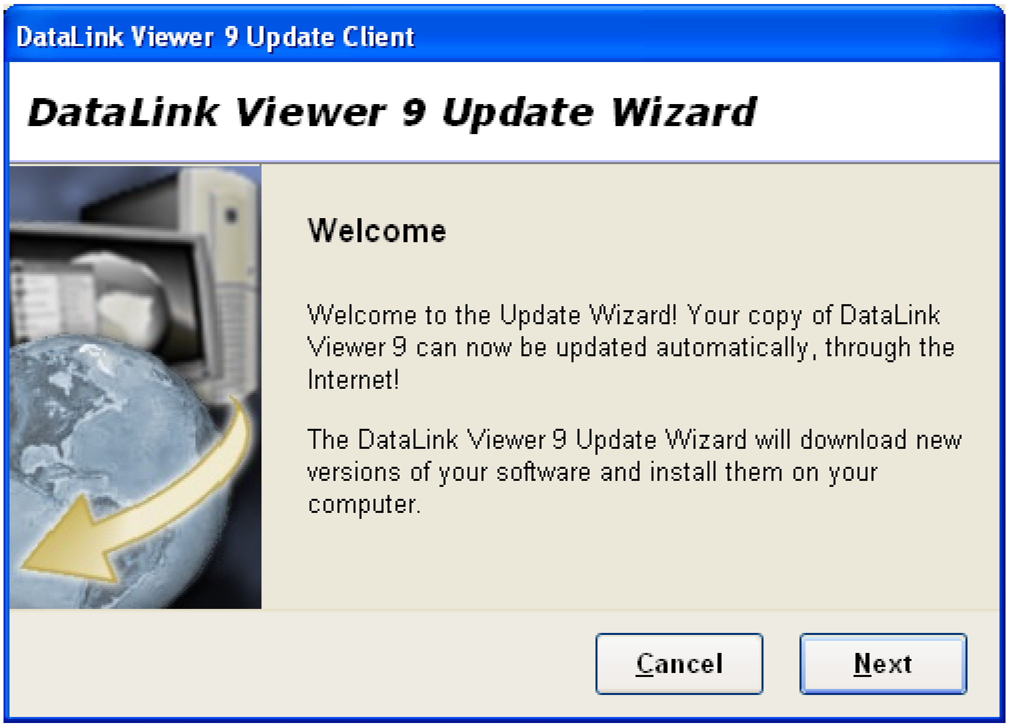


DataLink Viewer
Version 5.1002.90
www.MilletSoftware.com

Copyright 2002 Millet Software
By Ido Millet

E-mail: ido@MilletSoftware.com
Work Office: (814) 898-6262
Home Office: (814) 825-6009

When you click that button, an online update wizard detects if there are new versions and prompts you for a password to apply the patch (registered users with current annual support receive email notifications of such releases along with the necessary passwords).



DataLink Viewer 9 Update Wizard

Welcome

Welcome to the Update Wizard! Your copy of DataLink Viewer 9 can now be updated automatically, through the Internet!

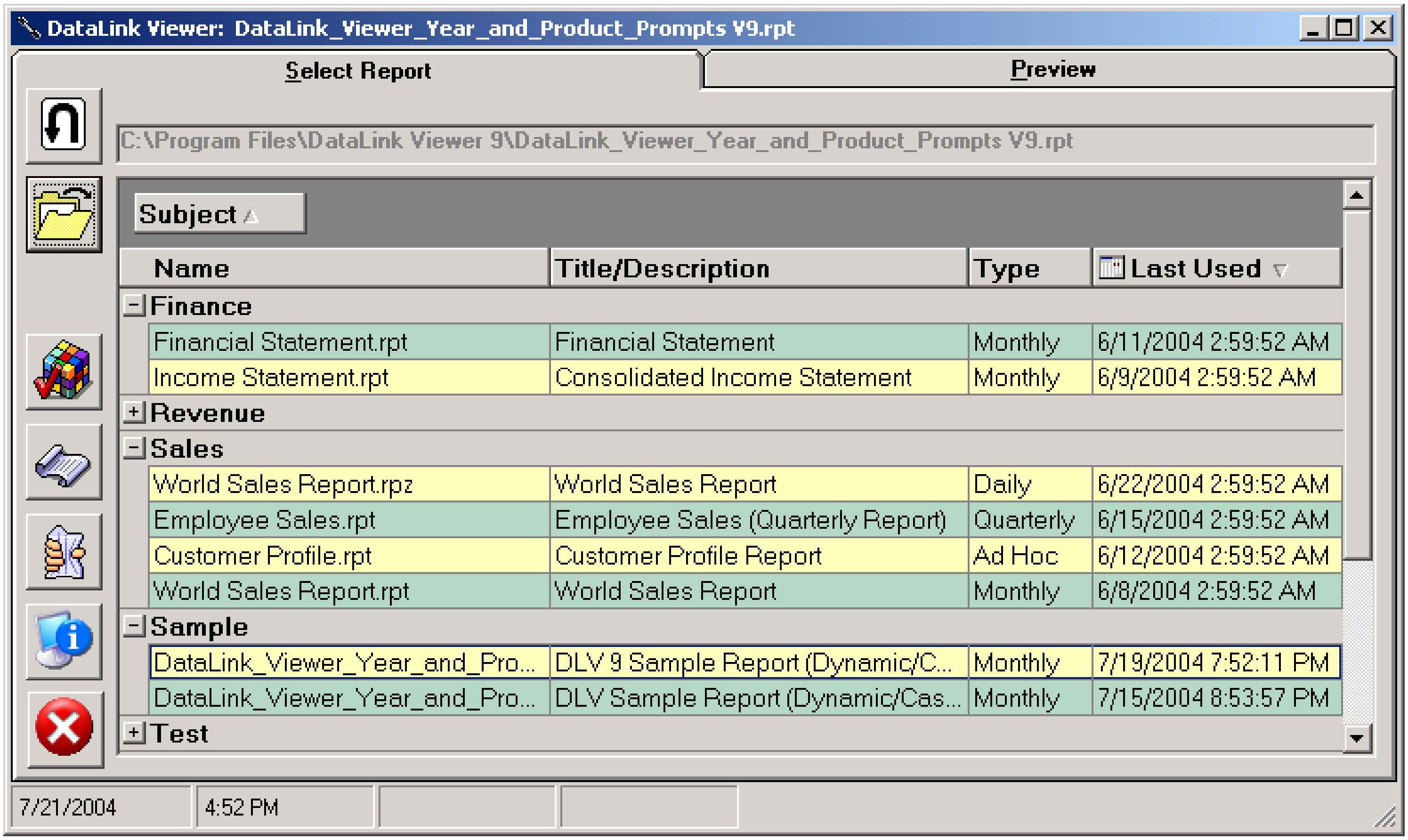
The DataLink Viewer 9 Update Wizard will download new versions of your software and install them on your computer.

Typical Use Scenarios

In a typical scenario, you would select a report to view by double-clicking it. The viewer would prompt you for any live or static parameters associated with that report. The viewer would then display the report with all the preview functionality available within Crystal (drilldown, tree view, export, zoom, print).
The report can also be invoked from a desktop shortcut, a batch file, or another program using DataLink Viewer's command line options.

User Interface

After starting *DataLink Viewer*, you would see a screen similar to this:


DataLink Viewer: DataLink_Viewer_Year_and_Product_Prompts V9.rpt
Select Report
Preview
Subject

C:\Program Files\DataLink Viewer 9\DataLink_Viewer_Year_and_Product_Prompts V9.rpt

Name	Title/Description	Type	Last Used
⊟ Finance			
Financial Statement.rpt	Financial Statement	Monthly	6/11/2004 2:59:52 AM
Income Statement.rpt	Consolidated Income Statement	Monthly	6/9/2004 2:59:52 AM
⊞ Revenue			
⊟ Sales			
World Sales Report.rpz	World Sales Report	Daily	6/22/2004 2:59:52 AM
Employee Sales.rpt	Employee Sales (Quarterly Report)	Quarterly	6/15/2004 2:59:52 AM
Customer Profile.rpt	Customer Profile Report	Ad Hoc	6/12/2004 2:59:52 AM
World Sales Report.rpt	World Sales Report	Monthly	6/8/2004 2:59:52 AM
⊟ Sample			
DataLink_Viewer_Year_and_Pro...	DLV 9 Sample Report (Dynamic/C...	Monthly	7/19/2004 7:52:11 PM
DataLink_Viewer_Year_and_Pro...	DLV Sample Report (Dynamic/Cas...	Monthly	7/15/2004 8:53:57 PM
⊞ Test			

7/21/2004 4:52 PM

Use the 📂 button to browse for and open a report for the first time. Previously selected reports are listed in a grid and can be launched by **double-clicking** or by **Right-Clicking** and selecting 'Preview Report' from the **popup menu**.

Use the ↻ button to reload a report (if changed and saved in Crystal).

To **resize the grid columns**, drag the borders between the column headers.

Use the button to access a dialog for setting various **options**. These options are maintained in the file **DataLink_Viewer.ini**

Use the button to open this **User Manual** as a pdf file.

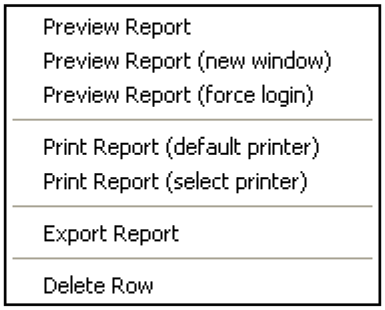
Use the button to **compress & encrypt RPT files into RPZ files**. This allows you to protect and hide your reports designs (either as an intellectual property issue or as a tech support issue). See the section on "Protecting Report Designs with rpz Files" for more detail on this feature.

Use the button to access a dialog with **Version (and system) information**. That dialog also has a button that allows you to check for software patches on my web site and install them online. This patching mechanism is very fast since the patches are typically very small (contain only file changes).

Use the button to **exit** DataLink Viewer.

<u>Right-Click Popup Menu:</u>

if you Right-Click a report row in the grid, the following popup menu is displayed:


Preview Report
Preview Report (new window)
Preview Report (force login)
Print Report (default printer)
Print Report (select printer)
Export Report
Delete Row

To **delete a report from the grid** (but not from the hard drive) select 'Delete Row' from the popup menu. The grid information in maintained in a plain text file (**ReportList.txt**).
The other options in the popup menu allow you to **launch a report to a new window**, **force a login dialog (allowing a choice of a different data source)**, and **printing or exporting the report without previewing it**.

When adding a report to the grid, DataLink Viewer populates the **title** and **subject** columns automatically if it finds that information in the **summary information** for the report.
You set that information for the .rpt file in Crystal (under the file menu).

<u>Customizing the Grid:</u> By dragging, clicking, or right-clicking the grid column headers you can apply various options such as grouping the report by any column(s), hiding/showing any columns, sorting, increasing/decreasing font size, etc.

Select a Report to View

Let's launch **DataLink_Viewer_Year_and_Product_Prompts.rpt** (installed under the **c:\Program Files\DataLink_Viewer\** directory in typical installations). You can do this by

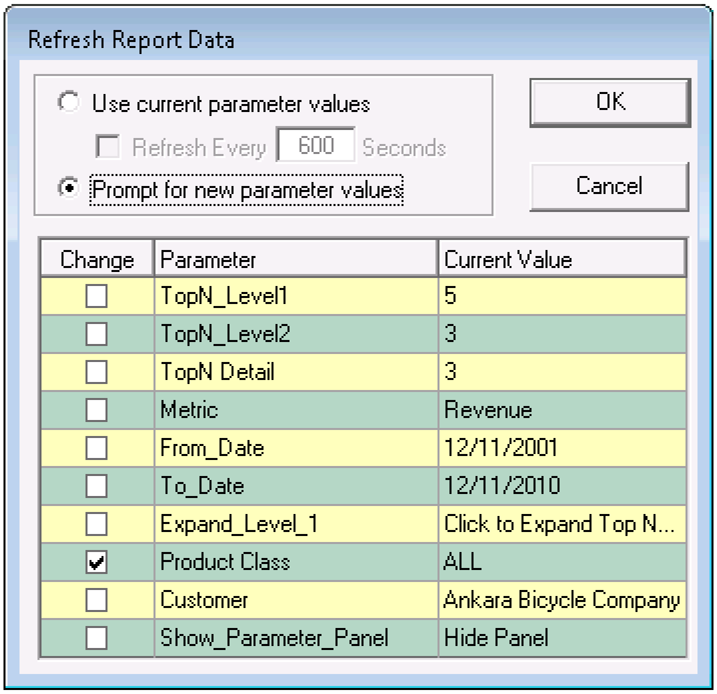
- Selecting that row and clicking the Preview Tab, or
- Right-Clicking that row and selecting Preview, or
- Double-clicking the row.

Note: use **DataLink_Viewer_Year_and_Product_Prompts V9.rpt or V11** if you are running the version of DataLink Viewer that uses Crystal 9 or XI runtime components. These sample reports assume that the Crystal Reports Extreme Sample Database (for 8.5, 9, or XI) is already installed on your PC.

Before displaying the report, DataLink Viewer would prompt you for any report parameters and logon information required by the report. **If the report was saved with data**, the viewer would ask you if you want to refresh the data or use the data saved with the report (in which case you would not be prompted for parameter values).

Selective Parameter Refresh

When refreshing a report that has parameters, you can select the parameters you wish to change. These choices are stored (in DataLink_Viewer.ini) for each report for reuse/ change at a later session.



Refresh Report Data
○ Use current parameter values
□ Refresh Every 600 Seconds
● Prompt for new parameter values
OK
Cancel

Change	Parameter	Current Value
□	TopN_Level1	5
□	TopN_Level2	3
□	TopN Detail	3
□	Metric	Revenue
□	From_Date	12/11/2001
□	To_Date	12/11/2010
□	Expand_Level_1	Click to Expand Top N...
☑	Product Class	ALL
□	Customer	Ankara Bicycle Company
□	Show_Parameter_Panel	Hide Panel

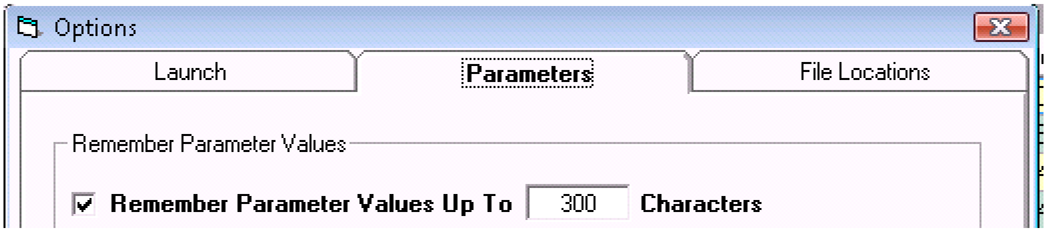
Notes:

- **Unlinked** subreport parameters participate in this process and are listed as [subreport name] -> Parameter Name.

- To remove **linked** subreport stored procedure parameters from this dialog, the parameter name must contains "_Linked".

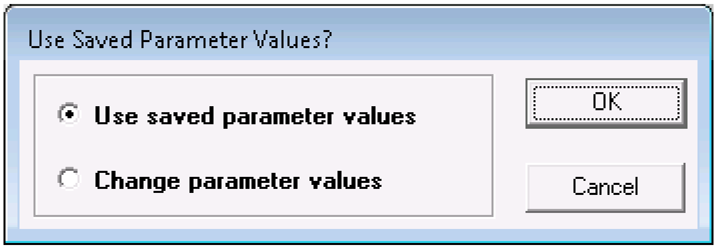
Remembering Parameter Values

By default, parameter values last used are saved to DataLink_Viewer.ini for reuse the next time you run the same report. Using the Options dialog and the checkbox shown below you can control whether that option is enabled. A parameter with more than 300 characters is not saved but you can adjust that number using the same area in the Options dialog:



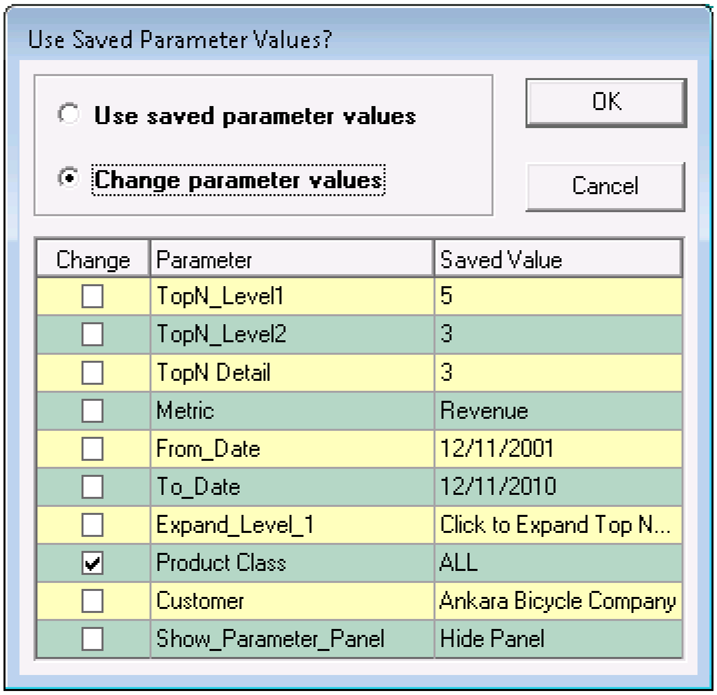
☑ Remember Parameter Values Up To 300 Characters

When you load a report that has saved parameter values, DataLink Viewer allows you to reuse these saved parameter values via a dialog similar to the Refresh Parameter Values dialog:



● Use saved parameter values

○ Change parameter values



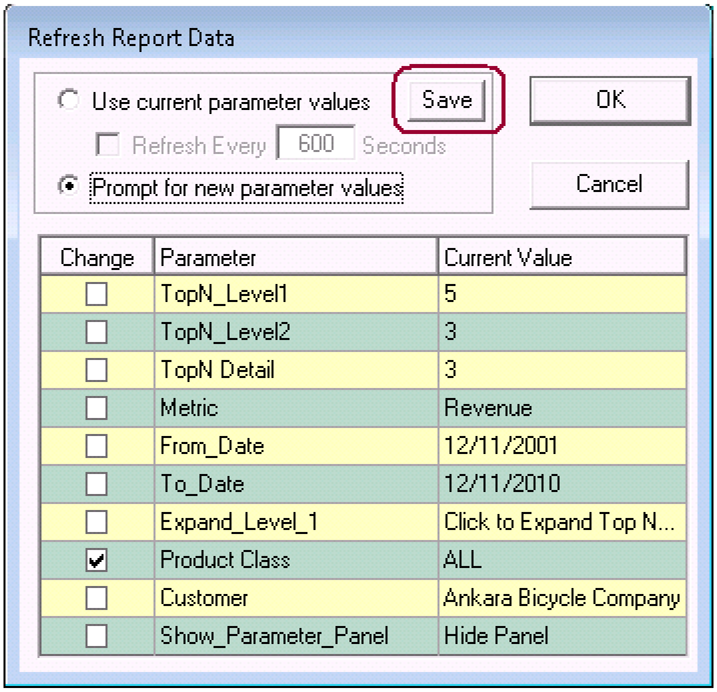
○ Use saved parameter values

● Change parameter values

Change	Parameter	Saved Value
□	TopN_Level1	5
□	TopN_Level2	3
□	TopN Detail	3
□	Metric	Revenue
□	From_Date	12/11/2001
□	To_Date	12/11/2010
□	Expand_Level_1	Click to Expand Top N...
☑	Product Class	ALL
□	Customer	Ankara Bicycle Company
□	Show_Parameter_Panel	Hide Panel

Save and Reuse Named Parameter Sets

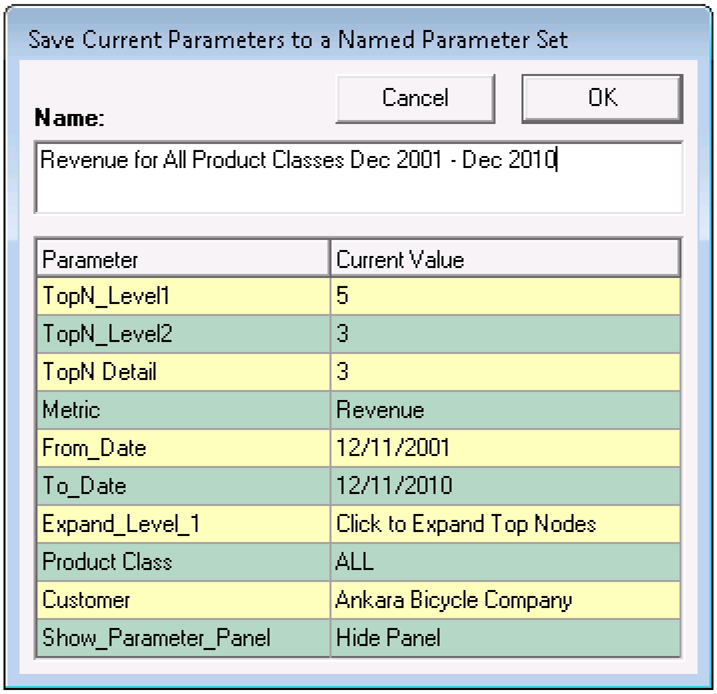
If you click the refresh button for a report that has at least 8 parameters (Options dialog allows you to change that number), a ***Save*** button becomes visible in the following dialog:



Refresh Report Data
○ Use current parameter values
Save
OK
□ Refresh Every 600 Seconds
● Prompt for new parameter values
Cancel

Change	Parameter	Current Value
□	TopN_Level1	5
□	TopN_Level2	3
□	TopN Detail	3
□	Metric	Revenue
□	From_Date	12/11/2001
□	To_Date	12/11/2010
□	Expand_Level_1	Click to Expand Top N...
☑	Product Class	ALL
□	Customer	Ankara Bicycle Company
□	Show_Parameter_Panel	Hide Panel

If you click ***Save***, the following dialog allows you to save the current parameters value set to DataLink_Viewer.ini under a unique name for this report.

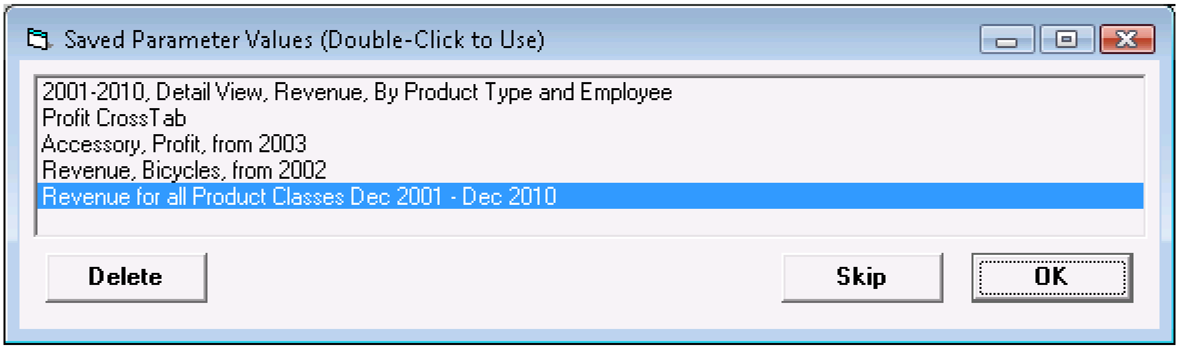


Name:

Revenue for All Product Classes Dec 2001 - Dec 2010

Parameter	Current Value
TopN_Level1	5
TopN_Level2	3
TopN Detail	3
Metric	Revenue
From_Date	12/11/2001
To_Date	12/11/2010
Expand_Level_1	Click to Expand Top Nodes
Product Class	ALL
Customer	Ankara Bicycle Company
Show_Parameter_Panel	Hide Panel

The next time you load this report, you would get a dialog that allows you select and reuse any of the saved named parameter sets for this report:



2001-2010, Detail View, Revenue, By Product Type and Employee
Profit CrossTab
Accessory, Profit, from 2003
Revenue, Bicycles, from 2002
Revenue for all Product Classes Dec 2001 - Dec 2010

Double-clicking any of the entries (or selecting an entry and clicking the OK button) would launch a dialog allowing you to reuse that set of saved parameter values or selectively change some of these saved parameter values.

This functionality was designed to address scenarios where reports with many parameters are used in one of several parameter patterns. By saving and naming these patterns, the user can reuse them.

Note: if you are a report developer, you can deliver these saved parameter patterns to a user machine by copying the [Named_Parameter_Sets] section in your DataLink_Viewer.ini file to the user’s DataLink_Viewer.ini file.

.

Dynamic & Cascading Parameters

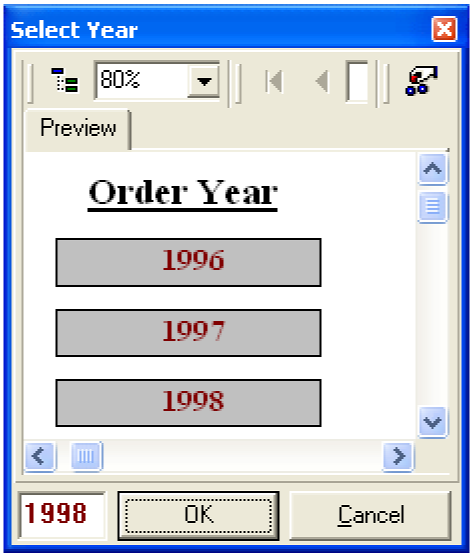
This section explains the special dynamic & cascading parameter functionality provided by DataLink Viewer. **When a parameter name matches a report name, DataLink Viewer uses that report as a parameter dialog**.

The first parameter in our sample report was named (by the report developer) as:
"Prompt_Order_Year.rpt"
DataLink Viewer detects that such a report exists and "links" to that report as the basis for prompting the user.

Note: if a prompting report doesn't exist, the static parameter dialog provided by Crystal will be used. ***DataLink Viewer* always tries to locate and use Live Prompts before using the static Crystal prompts.**

Respond to Single-Value Live Prompts

DataLink Viewer detects that the **Prompt_Order_Year.rpt** parameter was designed to accept only a single value, and hence it presents you with the linked report inside a dialog that allows you to click and select only a single value:



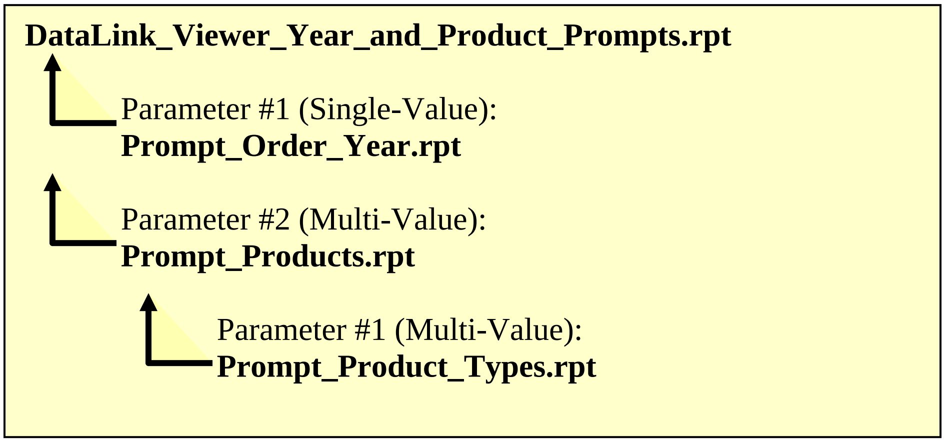
Note that by linking to a live prompt report, the designer of the report doesn't have to change the report design every year in order to add one more entry to the default value list of the static parameter. Instead, the prompt report dynamically presents all available years in the database.

Link back in a Hierarchy of Live Prompts

Once you select ***1997*** and hit OK, the viewer detects a second parameter in our sample report:
"**Prompt_Products.rpt**"
Again, ***DataLink Viewer*** detects that such a report exists and links to that report as the basis for prompting the User.

However, before presenting this report to the user, the viewer detects that the **Prompt_Products.rpt** itself is designed with a parameter of its own:
"**Prompt_Product_Types.rpt**"

What we have here is a ***hierarchy of links*** whereby the values selected by the user in response to the **Prompt_Product_Types.rpt** parameter restrict the values available for selection in the **Prompt_Products.rpt** parameter:



DataLink_Viewer_Year_and_Product_Prompts.rpt

Parameter #1 (Single-Value):
Prompt_Order_Year.rpt

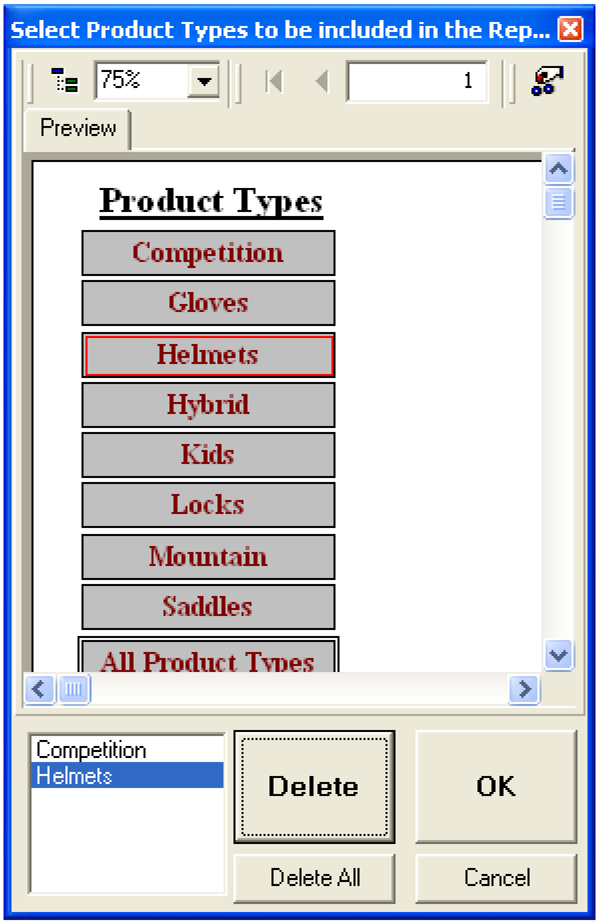
Parameter #2 (Multi-Value):
Prompt_Products.rpt

Parameter #1 (Multi-Value):
Prompt_Product_Types.rpt

DataLink Viewer **always starts from the bottom of the hierarchy for each parameter!**
Hence, we are going to be prompted first for Product Types before we select Products within those Product Types.

Respond to Multi-Value Live Prompts

DataLink Viewer detects that the parameter called **Prompt_Product_Types.rpt** was designed to accept multiple values, and hence it presents you with the linked report **Prompt_Product_Types.rpt** inside a dialog that allows you to click and select <u>multiple</u> values:

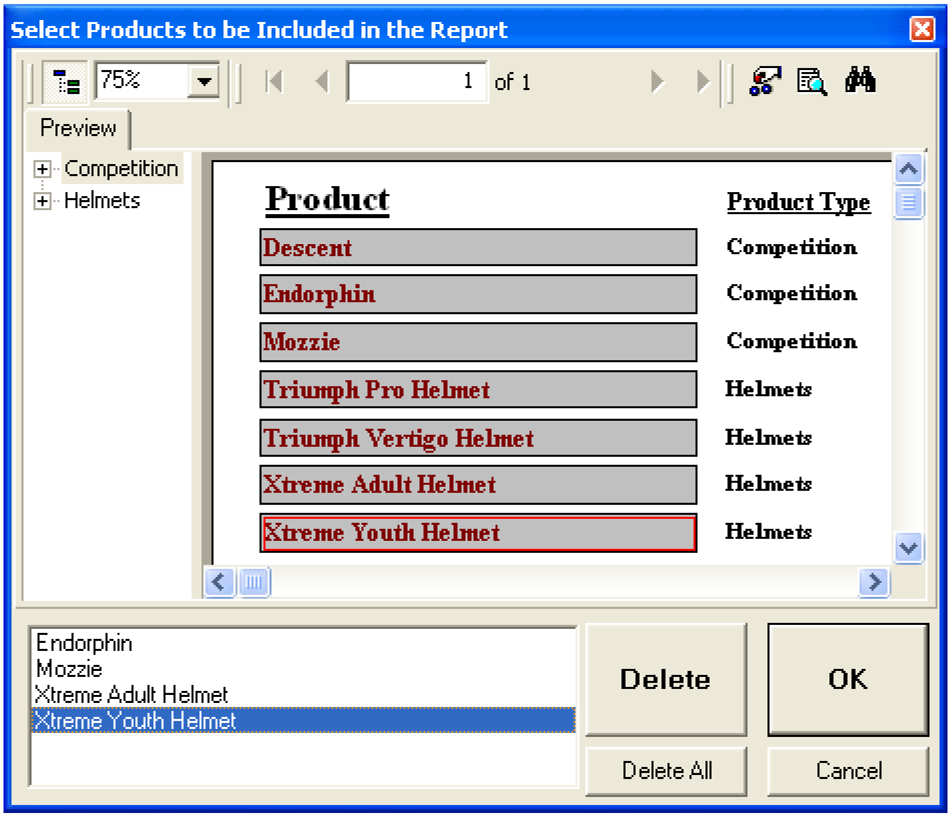


After clicking on *Competition* and Helmets to select those Product Types, click OK to progress to the Products selection within those Product Types.

Manage Display & Data Values for Live Parameters

In this particular situation, the **Product Type Name** is used for both **display value** (what gets shown to the user in the list of available values) as well as **data value** (what gets passed into Crystal as the parameter value). As demonstrated in the following step, *DataLink Viewer* allows you to present the user with meaningful product names as **display values** while passing Product IDs as the actual parameter **data value**.

Again, ***DataLink Viewer*** detects that the parameter called **Prompt_Products.rpt** was designed to accept multiple values, and hence it presents you with the linked report **Prompt_Products.rpt** inside a dialog that allows you to click and select multiple values. **Note that only Products within the Selected Product Types (in the previous step) are available for selection.** This demonstrates the power of **cascading dynamic parameters**:



Product	Product Type
Descent	Competition
Endorphin	Competition
Mozzie	Competition
Triumph Pro Helmet	Helmets
Triumph Vertigo Helmet	Helmets
Xtreme Adult Helmet	Helmets
Xtreme Youth Helmet	Helmets

Endorphin
Mozzie
Xtreme Adult Helmet
Xtreme Youth Helmet

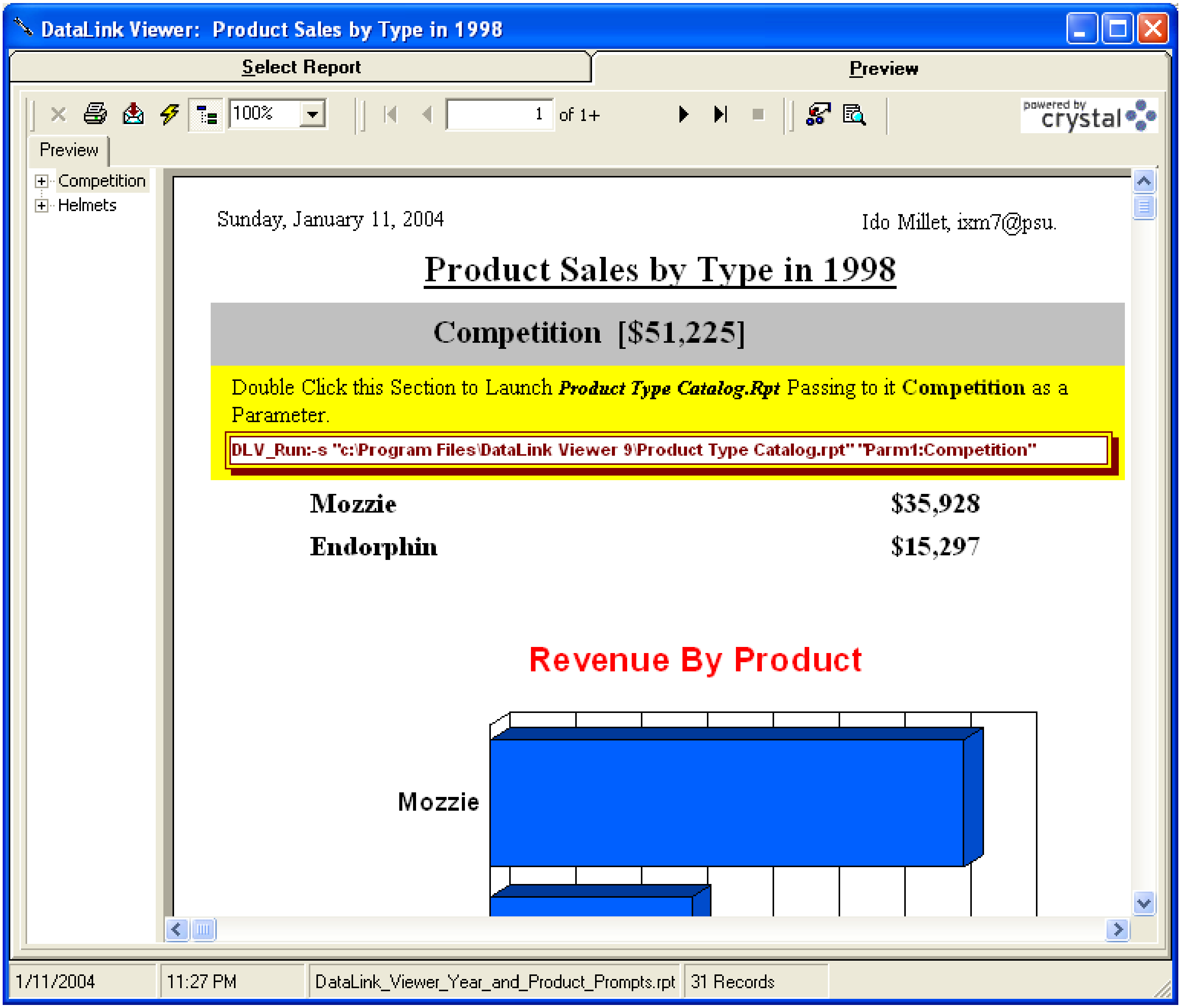
Notice that only the Product Name column is "active" and available for selection. The viewer allows you to click and select only formulas (and SQL expressions) that have the text "***Parameter***" as part of their names. This allows you to restrict the user to proper selections. It is also good practice to use color, font, and box effects to highlight the selectable column.

When you click on an "active" column (Parameter or Parameter_Data), the viewer always uses what you click on as the Display Value. However, if another column in the same row is named as "Parameter_Data", its value is used as the parameter Data Value. After clicking on some product names, move your cursor over these items in the list of selected Display Values. Notice that a popup text reflects the underlying Data Value (Product IDs for each product). Note: the @parameter_Data column's font was set to White to make it invisible.

You can **resize** the dynamic parameter dialogs, select a **zoom** level, and turn the **group tree** on or off for each dynamic report. **DataLink Viewer remembers these settings** and will apply them the next time the same dynamic parameter dialog (for the same dynamic parameter report) is loaded.

View the Report

When you click OK, DataLink Viewer uses the selected values for both the “Prompt_Year.rpt” and “Prompt_Products.rpt” parameters to display the requested report:



Sunday, January 11, 2004

Ido Millet, ixm7@psu.

Product Sales by Type in 1998

Competition [$51,225]

Double Click this Section to Launch ***Product Type Catalog.Rpt*** Passing to it **Competition** as a Parameter.

```
DLV_Run:-s "c:\Program Files\DataLink Viewer 9\Product Type Catalog.rpt" "Parm1:Competition"
```

Mozzie	**$35,928**
Endorphin	**$15,297**

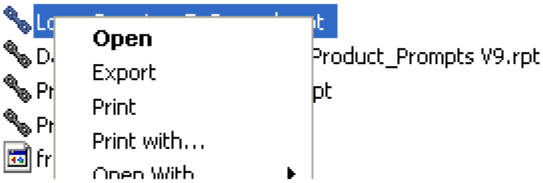
1/11/2004 | 11:27 PM | DataLink_Viewer_Year_and_Product_Prompts.rpt | 31 Records

Remember Parameter Values from a previous Session

If you refresh or reload the same report, the values you specified last time for each parameter are already selected, allowing you to accept (Hit Return or click OK) or replace them.

Launch Reports from File Explorer

DataLink Viewer allows you to launch a report from the Windows File Explorer by Right-Clicking a **.rpt** or **.rpz** file. The typical File Explorer menu provides the following extra options:



Open: launches a report to a Preview window.
Export: exports the report to a user selected export format and disk file.
Print: prints the report to the Default printer.
Print with…: prints the report to a user-selected printer.

These options allow users to invoke exporting or printing of a report without previewing it. This means users don't have to wait for the preview to complete before providing exporting or printing choices. Similar choices are available from the full user interface in DataLink Viewer (by right-clicking a report row in the 1st-tab grid). However, some users may prefer to use File Explorer as a launch mechanism without starting DataLink Viewer.

Note: DataLink Viewer associates these file explorer actions with **.rpt** and **.rpz** files only on PCs where these extensions are not already associated with other applications. This means that on a machine where Crystal Reports is already installed (where **.rpt** files are already associated with the Crystal application) you will not see these right-click menu options in File Explorer.

Launch Reports from Command Lines

DataLink Viewer allows you to launch a report from another application or from a shortcut using a command line.

The general structure of the command line is:
1) the full path and name of the **DataLink Viewer executable**

2) **–S** or **–V** as a processing flag indicating we are requesting a report to be **s**hown/**v**iewed
-S shows the report and ***allows selection of other reports*** via the Report Selection Tab
-V is a ***View Only*** option that ***allows only viewing of the specified report***
(the Report Selection Tab is not available to the user)

3) the **report path and file name**
4) Optional **Parameter value arguments** supplied as **“ParmN:Value”**
(where N is the position of the parameter in the parameter list within the Crystal report)
5) Optional **User ID** and **Password arguments**
6) Other optional arguments: Export, Printer, Print_Copies, ODBC_DSN**,** ODBC_DSN_From_To, Oracle_Server, Connect_To_SQLOLEDB, Auto_Refresh, ViewMode, Disable_Print_Button, Disable_Export_Button, Disable_Select_Button, Disable_Search_Button, Disable_Refresh_Button

Examples

1. Invoke viewing of the sample report while **passing 1997 as the 1st parameter value:**

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -s "C:\Program Files\DataLink Viewer 9\DataLink_Viewer_Year_and_Product_Prompts.rpt" "Parm1:1997"
```

2. Invoke viewing of a secure report while **passing User_ID and Passord:**

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -s "C:\Program Files\DataLink Viewer 9\Some Secure Report.rpt" "user_id:dba" "password:sql"
```

3. Invoke viewing of a report with only Preview tab (no Select Report tab) and specify an ODBC data source. This is useful in cases where you have multiple database with the same structure. You create several ODBC DSNs and can control via a command line which database is used:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\Program Files\DataLink Viewer 9\MyReport.rpt" "ODBC_DSN:Company1"
```

CALL DATALINK VIEWER FROM ANOTHER APPLICATION

4. Here's a code example of invoking DLV from a visual basic application with only a Preview tab (no Select Report tab) and specifying a parameter value. Note that double quotes are "escaped" by using
   `""`
   instead of
   `"`

   ```
   Dim RetVal
   ls_temp = "c:\Program Files\DataLink Viewer
   9\DataLink_Viewer_9.exe " & _
   "-v ""c:\temp\Lab2_Chase.rpt"" ""Parm1:800"""

   RetVal = Shell(ls_temp)
   ```

5. Here's a code example for invoking DLV from a vba event and dynamically setting the report path, name, and parameter value:

   ```
   Private Sub Combo0_AfterUpdate()
   Dim rs As Object
   Set rs = Me.Recordset.Clone
   rs.FindFirst "[txtReportName] = '" & Me![Combo0] & "'"
   If Not rs.EOF Then Me.Bookmark = rs.Bookmark

   Dim myreport As String
   Dim stAppName As String
   Dim myvalret As String
   ' me.fullrep is a field that concatenates the report path and name
   myreport = Me.fullrep
   myvalret = Str(MyCaseno)
   stAppName = "C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" & _
                  " -s """ & myreport & """ ""Parm1:" & myvalret & """"
   DoCmd.Close
   Call Shell(stAppName, 1)
   End Sub
   ```

Referring to Saved Encrypted Passwords

This section describes how to centralize and protect passwords by avoiding specifying them directly inside command line argument. Instead, you can name, encrypt and store the passwords inside DataLink_Viewer.ini. Besides protecting your passwords, this also allows you to change the passwords in one location, instead of in multiple command line arguments.

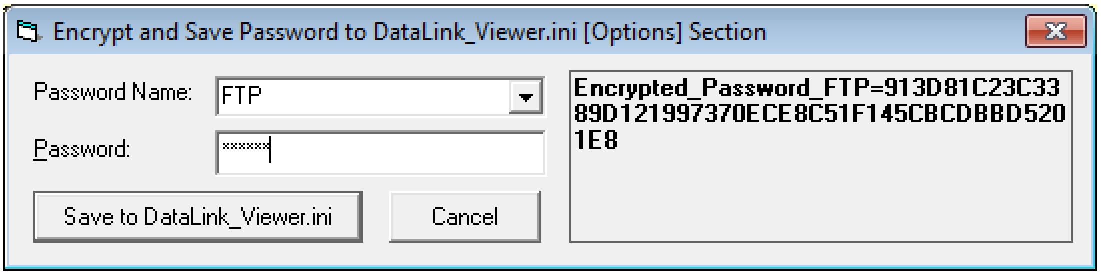
Instead of specifying a plain password in a command line such as:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\my.rpt"
"user_id:dba" "password:sesame"
```

You can refer to a saved and encrypted password like this:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\my.rpt"
"user_id:dba" "password:Encrypted_Password_FTP"
```

To save the encrypted password, go to the Launch tab in the Options dialog and click on the '***Encrypt & Save Password***' button. This provides the following dialog:


Encrypt and Save Password to DataLink_Viewer.ini [Options] Section
Save to DataLink_Viewer.ini
Cancel

Password Name: FTP

Password: ******

```
Encrypted_Password_FTP=913D81C23C33
89D121997370ECE8C51F145CBCDBBD520
1E8
```

The ini file entry is always named as "**Encrypted_Password_**" followed by the password name you specify. So in my case, the full name is **Encrypted_Password_FTP**. From that point on, I can refer to that password inside my command line arguments as ***Encrypted_Password_FTP*** and DataLink Viewer would make the appropriate substitution.

If you need to change a previously saved password, use the drop-down to select the previously saved password name and enter a new password.

Command Line Arguments for Parameter Values

Since parameter values can have different data types and structures (discrete, multi-value, range, mixed), this section explains how you can specify parameter values via command lines.

A numeric or string parameter would be specified as:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\Program
Files\DataLink Viewer 9\MyReport.rpt" "Parm1:1998"
```

A date parameter would be specified as:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\Program
Files\DataLink Viewer 9\MyReport.rpt" "Parm1:3/10/2003"
```

DataLink Viewer handles the data type conversion to match the parameter data type.

The syntax is constructed as the word “Parm”, followed by the number of the parameter (according to the order of parameters shown in Crystal), followed by a colon and the value.

Here’s how you would specify the 1st and 3rd parameter values:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\Program
Files\DataLink Viewer 9\MyReport.rpt" "Parm1:1998" "Parm3:Ido Millet"
```

All arguments must be **enclosed in double quotes** and **separated by a single space**.

Range and Multi-Value Parameters

DataLink Viewer supports all parameter types including multi-value, range, and mixed parameters. A multi-value discrete parameter value is specified as follows:

```
...  "Parm1:Competition:::Gloves:::Helmets"
```

A range parameter (in this case a date range) is specified as follows:

```
...  "Parm1:7/15/1996>>>7/15/2003>>>3"
```

The **3** at the end indicates the start and end points are included.
A **0** at the end would indicate the start and end points are NOT included.
A **2** at the end would indicate the start point is included and the end point is not.
A **1** at the end would indicate the start point is not included and the end point is.
Add 4 to these values if there is no Upper Bound.
Add 8 to these values if there is no Lower Bound.

Null Values

Null parameter values are specified in command lines by using the constant **[VC_NULL]**.
For example, to specify that the first parameter value is null, use:
“Parm1:**[VC_NULL]**”

Argument for Setting Formula Expressions

Note: this functionality is available only in DataLink Viewer 11 (XI R2) and 2011.

You can set formula expressions via a command line argument provided that:

- The formula name(s) must begin with a ^ character.
- Any double-quotes are specified as [dblq]
- Only main report formulas are targeted

The command line argument starts with **Set_Formulas1:** followed by pairs of **Formula Names** and **Formula Expressions**. The name is separated from the expression by >>>. Each pair is separated from the following one by ||| like this:

… **"Set_Formulas1:Name1>>>Expression1|||Name2>>>Expression2"**

For example, the following command line argument would set the first specified formula expression to the string “Ido” and the second formula to the numeric expression of 2 + 2:

… **"Set_Formulas1:{@^MyName}>>>[dblq]Ido[dblq]|||{@^TwoAndTwo}>>>2+2"**

Command Line Arguments for Triggering/Scheduling Exporting

Using the "**Export**" command line argument you can trigger, or even schedule, exporting for a report so the user doesn't need to preview the report or see the DataLink Viewer user interface.

INTERACTIVE USE

If you want the user to be prompted for the export format and file name, the syntax is constructed as the word "**Export**", followed by a colon and the word "**Dialog**". For example:

```
"C:\Program Files\DataLink Viewer 11\DataLink_Viewer_11.exe" -v "C:\temp\Report.rpt"
"user_id:dba" "password:sql" "Parm1:Gloves" "Export:Dialog"
```

FULLY-AUTOMATED EXPORTING

If you want to export the report with no user interaction, you can specify the export file name and export format like this:

```
"C:\Program Files\DataLink Viewer 11\DataLink_Viewer_11.exe" -v "C:\temp\Report.rpt"
"user_id:dba" "password:sql" "Parm1:Gloves" "Export:c:\temp\Labels.pdf||PDF"
```

Note that the export file name and export formats are separated by `‘||’`

Valid export formats include: `PDF, XLS, XLS Data Only, CSV, TTXV, CHSV, RECORDSTYLE, SEPARATEDVALUES, EXPLORER32EXTEND, HTML32, HTML40, DOC, RTF, EditableRTF, TXT, PaginatedTEXT, XML, and RPT.`

Notes:

- If the exported file **path is not specified** (for example, "Export:**test.pdf**||PDF"), the user gets prompted for a folder location. Last used export folder is remembered and reused.
- If **file name is left blank** (for example, "Export:||PDF") the user is prompted for a path and name for the exported file. If the user forgets to provide a file extension, it gets added automatically for most export formats. Last used export folder is remembered and reused.
- If a 3rd "SaveAs" element is included: "**Export:c:\temp\Labels.pdf||PDF||SaveAs**" then a Browse dialog allows the user to accept/change the export file name & path.
- If some parameter values are not specified, the user would be prompted for them.
- RPT exports the report to a Crystal Report format with saved data.
  A user can then open that report without needing access to the database.
- TTX provides export to Tab Separated Values.

SCHEDULING EXPORTING

Since you can trigger exporting using the Export command line argument, you can use the Windows Task Scheduler (or any other scheduler/application) to trigger exporting of the report. For detailed instructions on how to use the free windows task scheduler, see the section on "Scheduling" in my **Visual CUT User Manual** (Visual CUT provides much more powerful report scheduling options) at: www.milletsoftware.com/visualcutManual.htm

Command Line Arguments for Triggering Printing

Using the "**Printer**" command line argument you can trigger printing for a given report in an unattended mode (the user doesn't need to preview the report or see the DataLink Viewer user interface).

The syntax is constructed as the word “**Printer**”, followed by a colon and:

- the word "**Default**" if the report should be sent to the default printer, or
- the word "**Dialog**" if the user should be prompted to select a printer, or
- the **printer name**, if the report should be sent to a specific printer

Here’s how you would send a report to the **default printer**:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\Report.rpt"
"user_id:dba" "password:sql" "Parm1:Gloves" "Printer:Default"
```

Here’s how you would send a report to a **printer selected by the user**:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\Report.rpt"
"user_id:dba" "password:sql" "Parm1:Gloves" "Printer:Dialog"
```

Here’s how you would send a report to a **specified printer**:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\Report.rpt"
"user_id:dba" "password:sql" "Parm1:Gloves" "Printer:\\Srv1\Laser1"
```

Specifying Number of Copies

To control the number of printed copies, you can use a "**Print_Copies**" argument. The syntax is constructed as the word “**Print_Copies**”, followed by a colon and the number of desired copies. For example:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\Report.rpt"
"Printer:\\Srv1\Laser1" "Print_Copies:3"
```

Setting Custom Text for Each Print Copy

Using DataLink Viewer XI R2, in cases where **Print_Copies** is specified, each print copy can have custom text (such as 'Copy 1 of 2', 'Copy 2 of 2'). Simply place a Crystal formula called **{@Print_Copy_Template}** on the report canvas. For each printed copy, DataLink Viewer XI R2 would first update the text in that formula by replacing **[[N]]** with the copy number and **[[M]]** with the total number of copies. See this demo image.

Scheduling Printing

Since you can trigger printing using the Printer command line argument, you can use the Windows Task Scheduler (or any other scheduler/application) to trigger printing of the report. For detailed instructions on how to use the free windows task scheduler, see the section on "Scheduling" in my **Visual CUT User Manual** (Visual CUT provides much more powerful report scheduling options) at: www.milletsoftware.com/visualcutManual.htm

SCHEDULING PRINTING FOR MULTIPLE REPORTS

If you need to schedule printing of multiple reports at the same time, the best approach is to place all command lines into a single batch file. Then, schedule the batch file in the free Windows Task Scheduler as described above.

Launch a Report while Viewing another Report

DataLink Viewer allows you to view a report and launch another report into its own Viewer window or in the current Preview tab by double-clicking a report section containing a string formula specifying the report to launch, parameter values, and even User ID and Password.

The general structure of the String formula (no naming restrictions) is:
1) **DLV_Run:-v** (launch specified report in a **new Preview Only window**) or
DLV_Run:-s (launch specified report in a **new full DataLink Viewer window)** or
DLV_Run_Here:-s (launch the second report **within the current Preview Tab**)
2) the **report path and file name**
3) **Optional command line arguments** (parameter values, login information, etc.) as discussed in the "Launch Reports from Command Lines" section of this user manual.

Notes:

- The **User ID & Password values from the current report are automatically passed to the new report**. Specify these arguments only if you need to override these values.

- **-v** should be the **preferred option** when you launch a report into a new window because it removes the Select Report tab from the new window, simplifying what the user sees. This also increases the window space dedicated to showing the report.

Here are some examples of such String formulas:

```
// invoke a second report and load it into the Current Preview Tab
'DLV_Run_Here:-s "c:\directory\subdirectory\Report.rpt"

// launch into a Preview Only (-v) window and specify some parameter values:
"DLV_Run:-v ""c:\CR\My_Report.rpt"" ""Parm1:" + {Cust.CUST_N} +
""" ""Parm2:"  + {Prod.PROD_Code} +""""

// full DLV window (-s) and specify both login as well as Parameter information:
'DLV_Run:-s "Some_Report.rpt" "user_id:dba" "password:sql" "Parm1:1997"'
```

Note:

- Note: **No Text Wrapping Allowed**, so use very small font, and **to hide the formula you should use font colors, not the Suppress attribute**.

- When you need to embed double quotes into the result of a string formula, you must "escape" the double quotes with another double quote. Here are some variations on that theme:

…+ """" +	add a double quote.
"""…"	start the resulting string with a double quote
"…"""	end the resulting string with a double quote
"…""…"	insert a double quote into the middle of the resulting string

Create a User Interface for Selecting and Launching Reports

The ability described above (launching one report from another) allows you to very easily create a user interface that lets your users select and launch reports. The key steps are:

1. Create a REPORT Table with one record for each report. Typical columns would include: a) the path and report file name, b) a description of the report, c) report subject (Profit, Product Returns, Orders), d) report frequency (Daily, Weekly, Monthly), etc.

2. Create a "Report Selection" report listing these available reports with a formula in the detail section for launching the selected report. Note: you can design the main report with Parameters for listing only reports of certain subjects or certain frequencies. Of course, the REPORT table must store these classifications.

Let the user run the Report Selection report and launch the selected report by double-clicking it.

Embed Input from The User in the Command Line Call for Another Report

When launching another report by double-clicking a report section (as described above) you may wish to prompt the user for some input and embed that input into the command line. One scenario is a case where you wish to prompt the user to provide the value for a parameter. Another scenario may be a case where you wish to print labels for the product you are double-clicking but you want the user to be prompted for the number of copies to print.

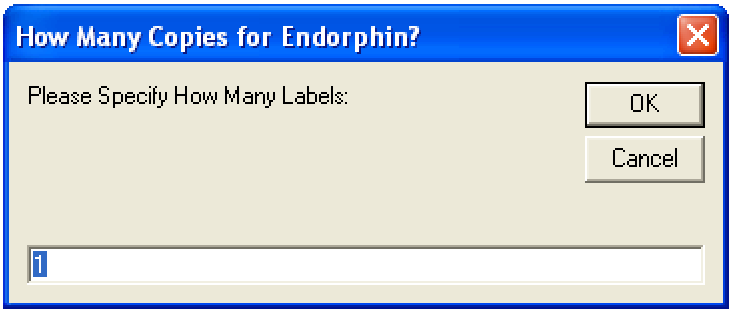
You can achieve this functionality by embedding within the command line a call to an InputBox function. DataLink Viewer would then replace the InputBox Call with the input from the user.

For example, the following formula would prompt the user for number of copies and then print the Product.rpt report for the double-clicked Product Name to the default printer with the number of copies specified by the user:

```
"DLV_Run:-v ""C:\temp\Product.rpt"" ""Parm1:" +
{Product.Product Name} + """" +
" ""Printer:Default""" +
" ""Print_Copies:#{InputBox||Please Specify How Many Labels:" & "||" &
"How Many Copies for " & {Product.Product Name} & "?" &
"||" & "1}#"""
```

The bold text is the InputBox call. This call must be enclosed by **#{… }#** and has 4 parts separated by "||":

1. InputBox (a keyword indicating this is a request for an InputBox dialog
2. The **Prompt** (the text shown within the InputBox dialog. If you need to specify line breaks within the prompt text, be sure to use **"vbCrLf"** (case sensitive), rather than Chr(10).
3. The **Title** (the text shown as the title of the InputBox dialog
4. The **Default** value (**1** in the example above)



How Many Copies for Endorphin?

Please Specify How Many Labels:

1

Notes:

1. On the report itself, the formula **must be displayed on a single line**, so **use small font**
2. to hide the formula you should use font colors, not the Suppress attribute

Launch another Application and Pass Parameters to It

DataLink Viewer lets you launch any other application and pass parameters to it by double-clicking a report section that contains a formula that statically or dynamically results in a string with the following structure:

'EXE_Run:Any_Program.exe:command line arguments'

Note:

- Note: **No Text Wrapping Allowed**, so use very small font, and **to hide the formula you should use font colors, not the Suppress attribute**.

- When you need to embed double quotes into the result of a string formula, you must "escape" the double quotes with another double quote. Here are some variations on that theme:

…+ """" +	add a double quote.
"""…"	start the resulting string with a double quote
"…"""	end the resulting string with a double quote
"…""…"	insert a double quote into the middle of the resulting string

Launching Reports in a New Window

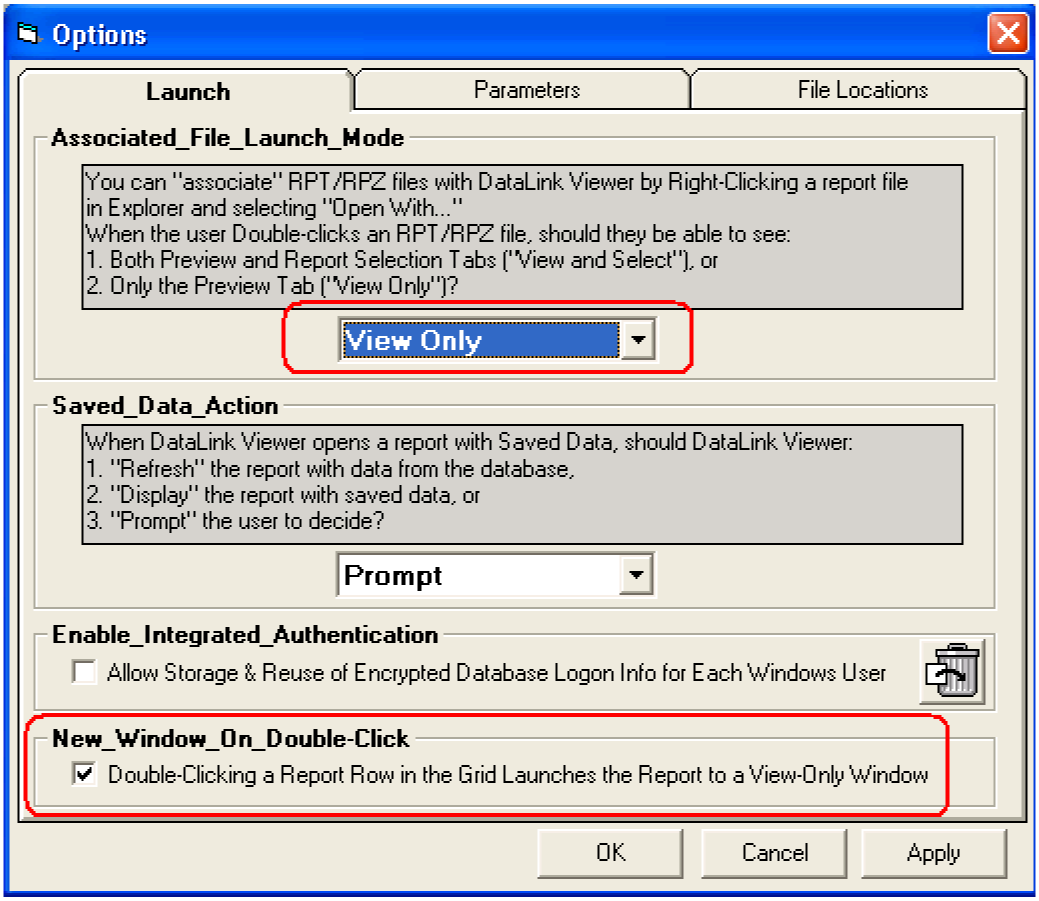
By default, the initial DataLink Viewer (DLV) window displays two tabs: ‘Select Reports’ and ‘Preview.’ You select a report from the first tab, and view it in the second tab. However, in many cases you may wish to **launch a report from the initial window into a new preview only window**. You may also wish to **remove the Preview tab from the initial window**. This section describes the various options for achieving this:

Launch a Report in a New Window

As described in the sections dealing with the command line api of DLV, you can launch reports in a Preview Only window by using –v instead of –s within the command line. However, this section deals with user interaction options for achieving the same thing. There are several ways to achieve this.

New Window On File Launch and/or On Double Click

The Launch tab in the DLV Options dialog allows you to direct DLV to launch reports into a View Only window when double-clicking an rpt file in File Explorer or when double-clicking a report row in the Select Report tab:



Associated_File_Launch_Mode

You can "associate" RPT/RPZ files with DataLink Viewer by Right-Clicking a report file in Explorer and selecting "Open With..."
When the user Double-clicks an RPT/RPZ file, should they be able to see:
1. Both Preview and Report Selection Tabs ("View and Select"), or
2. Only the Preview Tab ("View Only")?

View Only

Saved_Data_Action

When DataLink Viewer opens a report with Saved Data, should DataLink Viewer:
1. "Refresh" the report with data from the database,
2. "Display" the report with saved data, or
3. "Prompt" the user to decide?

Prompt

Enable_Integrated_Authentication

☐ Allow Storage & Reuse of Encrypted Database Logon Info for Each Windows User

New_Window_On_Double-Click

☑ Double-Clicking a Report Row in the Grid Launches the Report to a View-Only Window

Right-Click the Grid and Select Preview Report (new Window)

When you righ-click a report row in the Select Report tab, a popup menu provides you an option of **Preview Report (new window)**. If you click on that option, the report would launch in a new window.

Remove the Preview Tab from the Initial Window

If you wish to always use the initial DLV window to launch reports into new preview only windows, you may want to completely remove the Preview tab from the initial DLV window. You can achieve that by setting the following options in the **DataLink_Viewer.ini** file:

```
[Options]
Show_Preview_Tab=FALSE
```

Note: **this option takes effect only if you also use the Options dialog to turn on the option for Double-Clicking a Report Row in the Grid Launches the Report to a View Only Window**.

When these two options are set, DLV changes several things:

1. The initial DLV window shows only a Select Report tab
2. The right-click menu option of **Preview Report** is not shown. Instead, only the **Preview Report (new window)** option is visible.
3. The right-click menu option of **Preview Report (force login)** launches the report to a new window instead of to the Preview tab (which is not available).

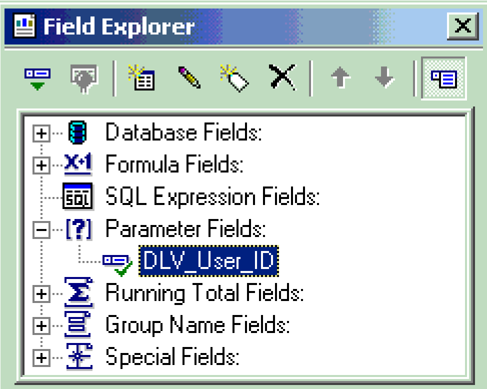
The end result is that the initial DLV window becomes just a launch pad to report previews.

Control Data Access according to PC Login

Assume you have several Sales Reps who need access only to their own sales information. DataLink Viewer can limit the data shown in a report according to who is logged in to the PC.

To achieve this functionality you need two elements:

1. The report must have a String parameter named "DLV_User_ID" as shown below. DataLink Viewer automatically sets the value of such a parameter to the User ID who is currently logged in to the PC.



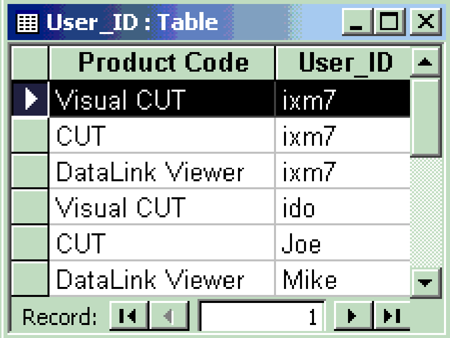
2. **The record selection formula must include a condition that restricts the data according to the User ID**. In most cases, this would be achieved by creating a new table in your database (or adding a new column to an existing table). For example, a new column in the Sales_Rep table (or a new table) can associate each Sales_Rep_ID with his/her User ID.

The record selection formula would include a condition such as:
{Sales_Rep.User_ID}={?DLV_User_ID}
This would ensure that as each Sales Rep logs into their PC, DataLink Viewer will show them only the sales records associated with their own Sales_Rep_ID.

Note: using a table, you can use multiple records to map the same User_ID to multiple areas of responsibility (Customer_IDs, Product_IDs, Region_IDs).

This allows you to give a single user (for example, a regional manager who should be able to see information for all her Sales Reps) permissions to view information that is related to more than one entity.

In the table example on the right, **ixm7** is allowed to view sales information for all three products, while **ido** is allowed to view sales information only for Visual CUT.



Product Code	User_ID
Visual CUT	ixm7
CUT	ixm7
DataLink Viewer	ixm7
Visual CUT	ido
CUT	Joe
DataLink Viewer	Mike

Load ini Values into Parameters

By naming parameters in a certain way, you can ask DataLink Viewer to automatically load their values from ini file entries.

One possible use may be for **reports where some parameters change only once per quarter**. Another case could be a situation where you develop and sell reports (probably .rpt files converted to .rpz files) to clients in a vertical market. These reports are designed to work against a known database structure, but **each client may need to customize the reports with text elements, conditional formatting, or record selection criteria without changing the report design**. Or perhaps you wish to **restrict use of your reports to only paying customers by providing a license code** that must match (using secret logic) the company name in the customer's database.

Instructions:
First, add to the [Options] section in **DataLink_Viewer.ini** (in the application folder) an entry with a **key name** (Company) and value (Millet Software) of your choice. For example:

```
[Options]
 …
Company=Millet Software
```

Then, add a **String Single-Value parameter** named **"DLV_INI_Option_KeyName"** to the report. DataLink Viewer automatically sets the value of such parameters to the value found for the Key Name under the [Options] section of DataLink_Viewer.ini. So, in the case above, the parameter value of **DLV_INI_Option_Company** would be set to "Millet Software".

Notes:

- Within DLV, the **user never gets prompted** for the value of such parameters.
- If a matching key name can't be found, the parameter gets the value of: "Failed INI Lookup"
- You can have as many parameters like this as you wish, each with a different key name.
- To pass such parameters to subreports, create a formula in the main report that simply returns the value of the parameter. Then, pass that formula as a link to the subreport.
- The ini file used by this process is always the MASTER ini file (the one in the DataLink Viewer application folder), even if you redirect to a DataLink_Viewer.ini at a different location. This allows you to enforce this type of parameters from a centralized location.

Securing Reports against Unauthorized Use

If you wish to secure your reports against unauthorized use, you can provide authorized users a License Key string for the ini entry. Within the report (later distributed to clients as an .rpz file) you design a record selection criterion that returns true only if the license key matches the company name in the database. As a simple example, you could check the number of characters in the license key is equal to the length of the actual company name (in the database) plus the number of R's in that company's name.

If you don't have access to the company name in the database, or if you wish to issue per-machine licenses, use the technique described in the next section.

Loading Machine Information into Parameters

DLV_MACHINE_NAME

If a report has a single-value string parameter called **DLV_Machine_Name**,
DataLink Viewer automatically sets the value of such a parameter to the Windows Machine Name on which DataLink Viewer is running.

DLV_REGISTERED_COMPANY

If a report has a single-value string parameter called **DLV_Registered_Company**,
DataLink Viewer automatically sets the value of such a parameter to the Company Name specified when the Windows operating system was installed..

In conjunction with the ability to load ini values into report parameters, this functionality allows report developers to distribute rpz files and restrict their use to only those where a license code matches the company to which Windows is registered.

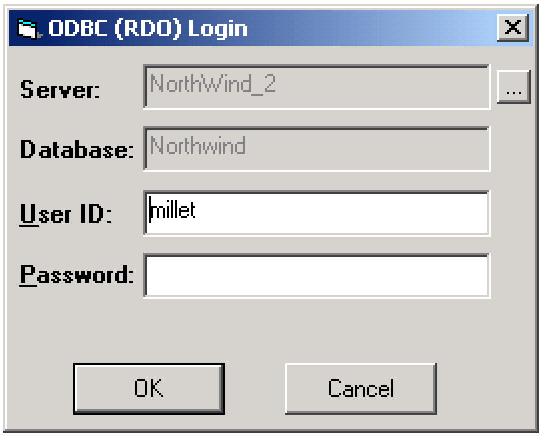
DLV_HD_SERIAL_N

If a report has a single-value string parameter called **DLV_HD_Serial_N**,
DataLink Viewer automatically sets the value of such a parameter to the serial number of the current hard drive.

In conjunction with the ability to load ini values into report parameters, this functionality allows report developers to distribute rpz files and restrict their use to only those where a license code matches the machine on which the report is running.

Database Choice Functionality

DataLink Viewer provides login dialogs to support any number of secure data sources in the main report as well as the subreports. Here is what the login dialog looks like:



ODBC (RDO) Login

Server: NorthWind_2

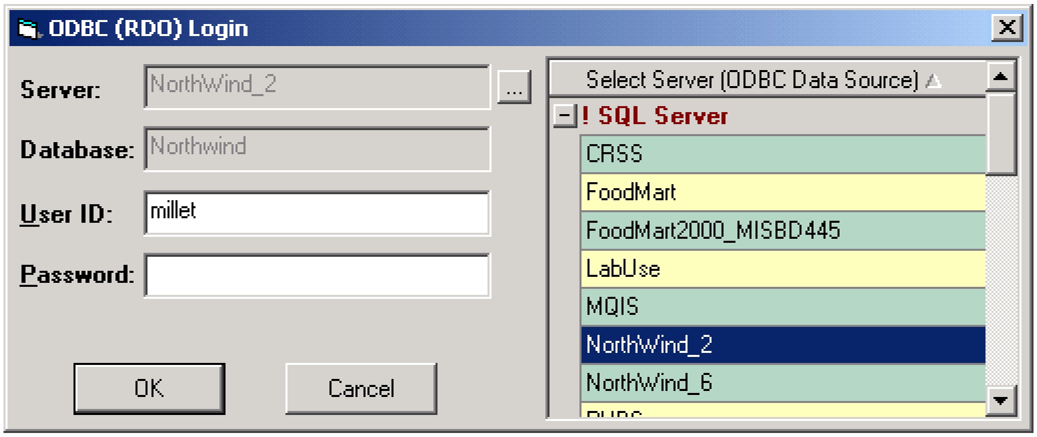
Database: Northwind

User ID: millet

Password:

Select Alternative ODBC Data Sources for the Same Report

When running reports that use ODBC, you can **select the ODBC data source to use**. Clicking a button to the right of the Server name, expands the display to include a listing of all available ODBC data sources (grouped by ODBC driver type). The original ODBC data source is initially selected and the driver group it belongs to is expanded (and prefixed with a "!"):



ODBC (RDO) Login

Server: NorthWind_2

Database: Northwind

User ID: millet

Password:

Select Server (ODBC Data Source)

- ! SQL Server
  - CRSS
  - FoodMart
  - FoodMart2000_MISBD445
  - LabUse
  - MQIS
  - NorthWind_2
  - NorthWind_6

Forcing the Login Dialog to Display to Allow Choice of ODBC DSN

If the data source is not password protected or if you are using the Integrated Authentication functionality of DataLink Viewer, the login dialog would not display. If you still wish to select a different ODBC DSN as the data source, you need to force the login dialog to display.

If you need to override the ODBC DSN only occasionally, the best option is to right-click the report row in the grid of previously opened reports and select the menu option of
Preview Report (force login).

If you need to always allow the user to select a different ODBC DSN, you can set the following options in DataLink_Viewer.ini under the [Options] section:
Always_Force_Login=TRUE
Display_DSN_List_in_ODBC_Login=True

Restricting the List of ODBC DSN Choices

The default ODBC DSN used by a report may belong to what you may consider to be a group of alternative ODBC DSNs. You may wish to restrict the list of ODBC DSNs options shown for such a report to only those that belong to that group. This can be accomplished by specifying an **ODBC_Groups** entry under a **[Login_Window]** section in the **DataLink_Viewer.ini** file.

Here is an example of such an entry:

```
[Login_Window]
ODBC_Groups={||Xtreme 9||Xtreme 11||} {||Northwind_2||NorthWind_6||Xtreme 9||}
```

Each ODBC Group is enclosed in { } brackets.
A single space separates the groups.
Each ODBC DSN is enclosed in a || delimiter (including at the start and end of the group.

report that by default uses an ODBC DSN that is NOT in any of the specified groups will experience no restrictions in the choice of alternative ODBC DSNs.

A report that by default uses an ODBC DSN that is in only one of the specified groups will be restricted to a choice among the DSNs of that group.

A report that by default uses an ODBC DSN that is in more than one of the specified groups will be restricted to a choice among the combined list of DSNs from all these matching groups.

Using Command Line Arguments to Specify the ODBC DSN

In some cases, you may want to use the same report to connect to different data sources, such as a **testing** or **production** server. While each report stores connection properties for only one default ODBC Data Source Name (DSN), DataLink Viewer allows you to use command line arguments to specify a different ODBC DSN.

Note: You can combine this functionality with the **user_id** & **password** command line arguments described in a previous section.

The command line argument structure is as follows:

Selecting an Alternative ODBC Data Source

… "ODBC_DSN:Data Source Name"
or
… "ODBC_DSN_From_To:Old_DSN1>>New_DSN1||Old_DSN2>>New_DSN2"

The **ODBC_DSN** argument overrides all ODBC DSNs used in the report by the new DSN
The **ODBC_DSN_From_To** argument overrides only tables that use the old DSN.

Notes:

1. You can combine this functionality with the **user_id** & **password** command line arguments described in a previous section.

2. The target **Data Source Name** must exist on the PC running Visual CUT. If it doesn't, Visual CUT reverts back to running the report against the ODBC DSN the report was designed for (a warning message is added to the failure.log file).

3. **ODBC_DSN_From_To** support **multiple pairs** separated by "**||**" as shown in the example above. This addresses scenarios where a report uses multiple ODBC DSNs (e.g. when subreports use different DSNs).

4. The **ODBC_DSN_From_To** is not available in the Crystal 8.5 version of DataLink Viewer.

5. **IF ODBC_DSN_From_To** is spelled as **ODBC_DSN_FROM_To (FROM in all upper case)**, if none of the tables in the report used the DSN specified as FROM, an error message is displayed.

Overriding the Database Specified in the Report or ODBC DSN

For ODBC data sources, you can enter a database name into the login dialog if you wish to override the database specified in the ODBC data source or in the report itself. This is useful for situations where the same database (e.g., MS SQL Server) contains multiple databases each with the same table structure. If the number of such databases is large, creating a dedicated ODBC DSN for each and using the select ODBC DSN functionality may be too tedious. Instead, you can directly type in the database name in the login dialog.

To enable database name input into the database field in the login dialog, you need to set the **Override_ODBC_DSN_Database** entry under the [Options] section in the DataLink_Viewer.ini file to TRUE, like this:

```
[Options]
Override_ODBC_DSN_Database=TRUE
```

Note that if that option is set to TRUE, the database specified in the ODBC DSN can no longer override the database specified in the report (if the user doesn't type in a Database, the specified Database remains the one used in the report.

Note: This functionality is not available in the Crystal 8.5 version of DataLink Viewer.

Stripping Table Qualifiers when Connecting to a Different Database

The SQL statement generated by Crystal frequently contains not only table names, but also the database name used at the time the report was designed (Database.Table or Database.Owner.Table). Such table qualifiers can frustrate attempts to run the report against a different database.

DataLink Viewer supports selection of a new ODBC data source even when the database name is different from that used in the original ODBC DSN. To ensure this works even when the table names in your report are fully qualified, set the following line under the **[Options]** section in **DataLink_Viewer.ini** to True:

```
Strip_Table_Qualifiers=True
```

Overriding the Server in Native Oracle Connection

When a report uses a **native connection** to Oracle, you can **edit the Server name** in the login dialog and run the report against a different server (rather than the one the report was designed against).

Alternatively, if you are launching a report from a command line, you can override the Oracle server name by using the "Oracle_Server:" command line argument. For example:

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\test.rpt"
"user_id:dba" "password:sql" "Oracle_Server:Server2"
```

Note: This functionality is not available in the Crystal 8.5 version of DataLink Viewer.

Selecting an Alternative SQL Server – OLE DB Data Source

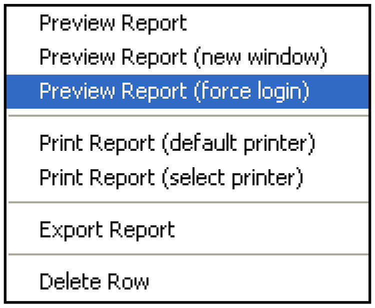
… "Connect_To_SQLOLEDB:DataSource>>InitialCatalog>>Integrated_Auth"

The parameters (after the ":") are separated by a ">>" and are as follows:
1. **DataSource** is the **Server Name** the report should connect to.
2. **InitialCatalog** is the **Database Name** within the given server.
3. **Integrated_Auth** is a **True** or **False** argument indicating if Microsoft SQL Server Integrated Authentication should be used (if TRUE, user_id & password are ignored).

Note: This functionality is not available in the Crystal 8.5 version of DataLink Viewer.

Forced Login

Right-clicking the report list provides a **Preview Report (force login)** popup menu option for forcing a login window before previewing the report:


Preview Report
Preview Report (new window)
Preview Report (force login)
Print Report (default printer)
Print Report (select printer)
Export Report
Delete Row

This is useful when:

- a previous report connected to one ODBC data source and you wish to run the same (or another) report against another ODBC data source.
- you wish to login to the same data source under a different user id, without restarting DataLink Viewer.
- You wish to avoid the default login information provided by Integrated Authentication (without bothering to go into the Options dialog and turning Integrated Authentication off)

Note: for some ODBC data sources (Crystal Commands), there are some scenarios where the only way to avoid connecting to a previously opened connection is to close and reopen DataLink Viewer.

Selecting Folder Location for FoxPro DBF Files (MasterBuilder)

The CR 9 (and above) runtime components versions of DataLink Viewer support dynamic selection of data folders for reports using the Visual FoxPro ODBC driver for a File DSN (using "Free Tables", which are dbf files under a given folder).

This functionality was originally developed to support ***Intuit Master Builder*** company reports where each company's data is located in a different folder on the same PC.

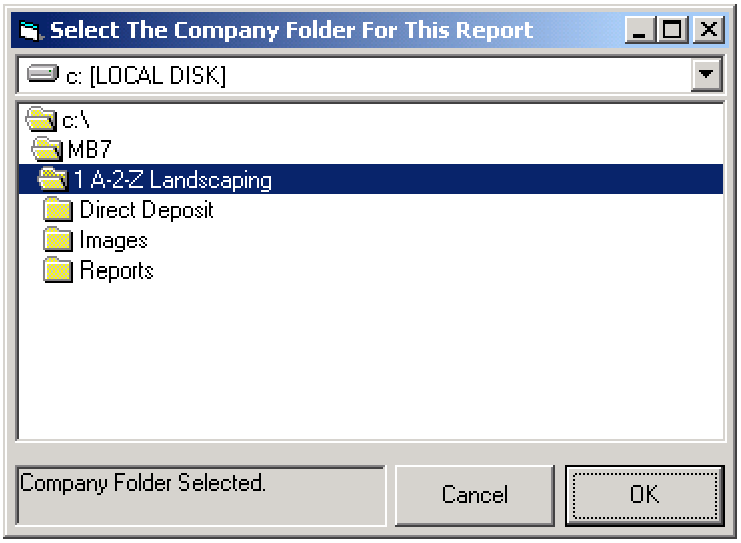
Two sections in the DataLink_Viewer.ini file control this functionality:

```
-----------------------------------------------------------------
[Folder_Selection_DSNs]
Visual_FoxPro_DBF=||MB7 Data.dsn||MB Data.dsn||

[MB7 Data.dsn]
Folder_Selection_Must_Contain=company.dbf
Enable_Folder_Selection=TRUE
-----------------------------------------------------------------
```

The 1st section specifies which File DSNs call for this new functionality. The File DSNs are specified exactly as they are named in the report itself. They must be enclosed in "||" as delimiters (even for the first, last, or only entry)

The 2nd section specifies, in the case of Master Builder data, that users should be allowed to select only folders that contain a **company.dbf** file. That section also allows users to Enable or Disable dynamic folder selection.



c: [LOCAL DISK]

Company Folder Selected.

Changing Folder Location for Access/Excel/Paradox Files

Note: this functionality is not available in the 8.5 version of DataLink Viewer

If your report uses the native connection to MS Access or Excel files, you can control the location of the database files using the following section in DataLink_Viewer.ini

```
-------------------------------------------------------------------
[Database_Path_Selection]
Paths = C:\Old\xtreme.mdb>>C:\New\xtreme.mdb||C:\a\test.mdb>>?
// for Paradox specify only path to DB files: Path = C:\Old\>>c:\New\
-------------------------------------------------------------------
```

Note: you can also use a **"Database_Path_Selection:… " command line argument.**

As demonstrated above, the Paths entry may contain multiple pairs of old>>new paths.
Each pair specifies the **old path** followed by a "**>>**" separator, followed by the new path.
The pairs are separated by "||"

If the new path is blank, or if it contains just a question mark, DataLink Viewer will prompt the user to select a new path when a report using the old path is first used.

If the new path has one question mark (**?**) followed by a valid path, that path will be the default location in the dialog asking the user to select a path.

If instead of a single question mark, the new path starts with a double question mark (**??**), DataLink Viewer will always prompt the user to select a path each time a report using the old path runs. The user choice will be added after the **??** and will become the new default value in the path selection dialog.

If the report uses a database file that can't be found on the machine, DataLink Viewer prompts the user to select a valid location, creates the **[Database_Path_Selection]** section (if it doesn't already exist) and creates/adds the pair information to the **Paths** entry.

These options are designed to address typical deployment scenarios when a report developer sells reports to users who may have their data source location at a different folder than the one used when the reports were developed.

USING COMMAND LINE ARGUMENT TO AVOID PATH PROMPT

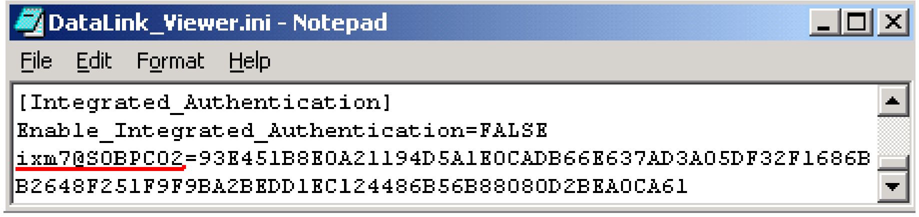
By default, pairs with two question marks before the new default path (old_path>>**??**new_path) cause a prompt each time the old path is encountered in a report. You may wish to avoid such repeated prompts when launching one report from within another report or when scheduling printing. The following command line argument will cause DataLink Viewer to ignore the two question marks and simply use the new_path.

… "DB_Path_Use_Default:TRUE"

Integrated Authentication

Integrated Authentication ("Remember Me")

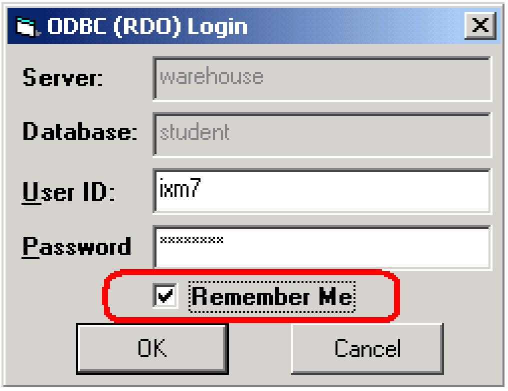
DataLink Viewer allows users to **avoid repeated login prompts to databases**. The database login information is stored, highly encrypted, inside **DataLink_Viewer.ini** as shown below:



```
[Integrated_Authentication]
Enable_Integrated_Authentication=FALSE
ixm7@SOBPC02=93E451B8E0A21194D5A1E0CADB66E637AD3A05DF32F1686B
B2648F251F9F9BA2BEDD1EC124486B56B88080D2BEA0CA61
```

The security of login information is maintained not only by storing it in **encrypted** format, but also by storing it with an internal identification of which **Windows User & PC** this login information belongs to. **DataLink Viewer uses this encrypted database login information only after checking that the same Windows user is running from the same PC.** In other words, users cannot break the login security by attempting to copy and paste the encrypted information to their own ini file entry).

For example, in the example shown above integrated authentication has been enabled. This can be done via a checkbox in the Option dialog (Launch Tab). The user (**ixm7**) then logged in to a secure database by providing a **database user id & password**. The user turned on the "**Rememebr Me**" option (visible only when integrated authentication is enabled):



Server: warehouse

Database: student

User ID: ixm7

Password: *********

☑ Remember Me

The information was then saved for the **ixm7** user, running on the **SOBPC02** PC as shown in the ini file above. **From that point on, the same user (ixm7), once logged to the same PC (SOBPC02), doesn't need to manually login to the same data source.**

The same user can turn on the "Remember Me" option for **unlimited number of secure data sources** and the information for all of them would be maintained inside the encrypted entry. The Options dialog allows each user to delete their own integrated authentication information by clicking a button.

Shared Machine Authentication

This section describes how one user can elect to share their integrated authentication information with any other user who successfully logged in to the same machine.

Step 1: First, add the entry in bold to the DataLink_Viewer.ini file:

```
---------------------------------------------
[Integrated_Authentication]
Enable_Integrated_Authentication=TRUE
ixm7@SOBPC02=18CB9CAE3D8BF3484000123301DD155638EAD4AD6B4D622C04DF125B7404BCBC4717A5DA2FBCD94314A7CE63BBF7357E
Enable_Shared_Machine_Authentication=TRUE
---------------------------------------------
```

Step 2: Then, as the "key" user who will share integrated authentication, run a report and get to a login dialog. Note: if you already have integrated authentication for yourself, use Forced Login. Alternatively, discard your integrated authentication (using the Trash Can button in the Options dialog). Make sure you turn on the Remember Me option in the login dialog, just like setting up integrated authentication for yourself.

Step 3: Because of the **Enable_Shared_Machine_Authentication=TRUE**
you would get a dialog asking you if you wish to share your integrated authentication functionality with other users who logged in to the same machine. This ensures a user can't be tricked into sharing integrated information without their knowledge and expressed consent.
Click YES.
If you then open the DataLink_Viewer.ini file you would notice this process generated a new entry (in bold):

```
---------------------------------------------
[Integrated_Authentication]
Enable_Integrated_Authentication=TRUE
ixm7@SOBPC02=18CB9CAE3D8BF3484000123301DD155638EAD4AD6B4D622C04DF125B7404BCBC4717A5DA2FBCD94314A7CE63BBF7357E
Enable_Shared_Machine_Authentication=TRUE
Shared_Machine_Authentication=C8B6CD373D5D5ADDC6FE9F379521C7318FA297CB595B992C
---------------------------------------------
```

That entry, in my particular case, points to the **ixm7@SOBPC02** integrated authentication entry.

Notes:

1. The **Enable_Integrated_Authentication** (True or False) setting is **always taken from the "user" ini file.** This allows the user to turn it off if they wish. For example, they may wish to switch between ODBC Data Sources by using the special option in the login dialog.

2. The **Enable_Shared_Machine_Authentication** and **Shared_Machine_Authentication** information is **always taken from the "master" ini file**. This allows an administrator to enable/disable and change the shared machine authentication for all users in one central file.

3. The regular integrated authentication information (for a **user@machine** entry that matches the logged in user id and the machine id) is always taken from the user ini file, if it exists.

4. **There can be only one shared machine authentication entry**. If you need to change to another administrator, delete the **Shared_Machine_Authentication** line and let the new administrator go through the Remember Me and dialog step.

5. If the administrator adds more data sources and login information to their integrated authentication information, all the machine users would have access to the new data sources. This is because the entry is really a "pointer" to whatever that administrator has accumulated in their entry (ixm7@SOBPC02 in my case).

Shared Secret Password with Windows User IDs

DataLink Viewer has special functionality allowing an administrator to set/change a secret global password for all users who will then be authenticated to the database using their own windows user id and the secret global password.

If you are in the rare situation where you need to use this functionality, contact Millet Software and, if your use scenario matches this functionality, you will receive detailed instructions.

Monitoring DataLink Viewer Use

DataLink Viewer allows you to monitor the use and performance of reports across the organization via logging to an ODBC table or to a simple text file. Logging to ODBC is a more secure and powerful approach and is described first.

Record Processing to an ODBC Database

To log processing to an ODBC database, you must create a table called DLV_Log in the target database. Email to me if you'd like to receive a sample MS Access database with the required DLV_Log table.

Below are examples of data structures for MS Access and MS SQL Server. However, those are just examples. You can use such a table in any other ODBC aware database (Oracle, DB2, Sybase, etc.).

TYPICAL USE:

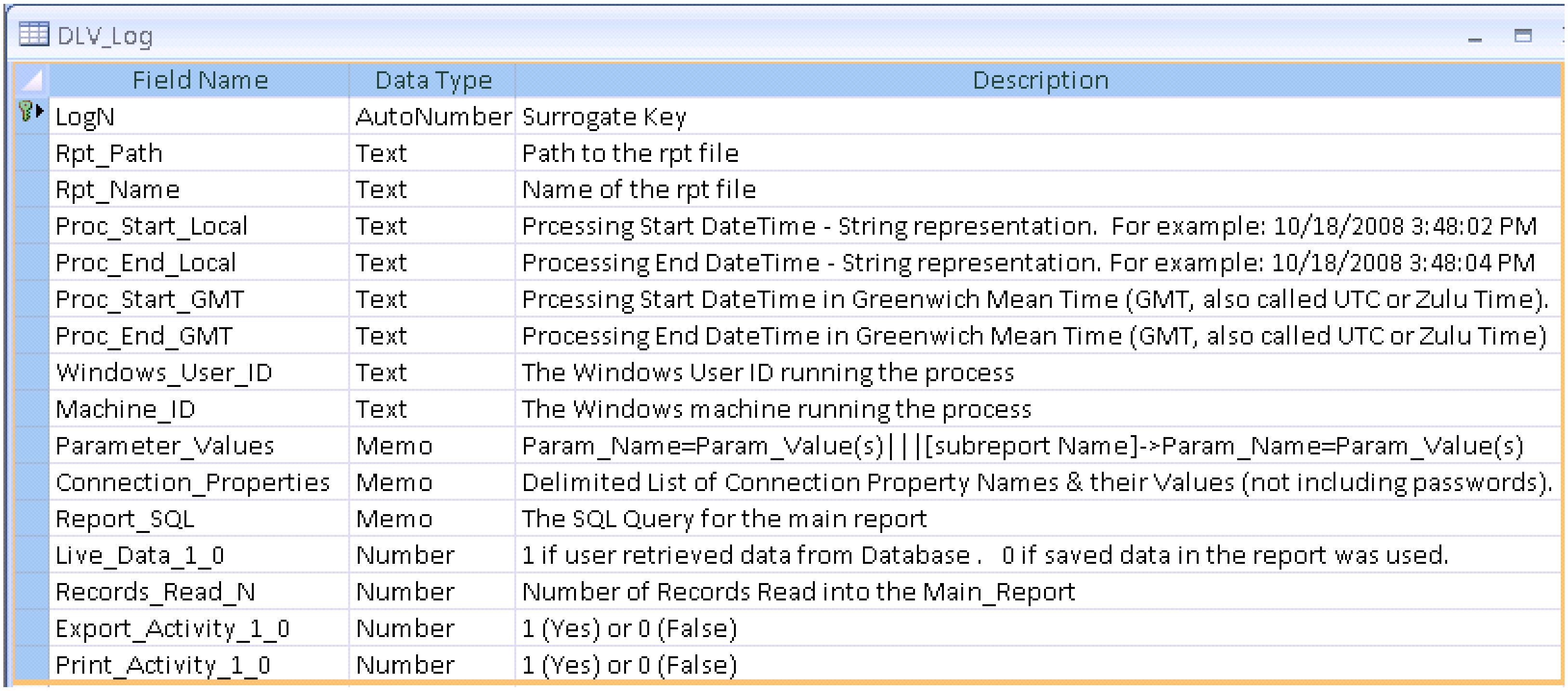
The DLV log table can be used to:

1. Monitor report use
2. Monitor report performance (using the start & end time information)
3. Monitor what employees access/print/export what data. This can be useful for addressing data privacy and protection concerns and requirements (e.g., HIPAA)

If there is a need to monitor this log for exception situations, you can use Visual CUT (another Millet Software tool) to schedule exception reports and email alerts against this table.

MS Access Data Structure

Here is the table structure required for this table, when implemented under MS Access (along with comments explaining what information is recorded):



DLV_Log

Field Name	Data Type	Description
LogN	AutoNumber	Surrogate Key
Rpt_Path	Text	Path to the rpt file
Rpt_Name	Text	Name of the rpt file
Proc_Start_Local	Text	Prcessing Start DateTime - String representation. For example: 10/18/2008 3:48:02 PM
Proc_End_Local	Text	Processing End DateTime - String representation. For example: 10/18/2008 3:48:04 PM
Proc_Start_GMT	Text	Prcessing Start DateTime in Greenwich Mean Time (GMT, also called UTC or Zulu Time).
Proc_End_GMT	Text	Processing End DateTime in Greenwich Mean Time (GMT, also called UTC or Zulu Time)
Windows_User_ID	Text	The Windows User ID running the process
Machine_ID	Text	The Windows machine running the process
Parameter_Values	Memo	Param_Name=Param_Value(s)|||[subreport Name]->Param_Name=Param_Value(s)
Connection_Properties	Memo	Delimited List of Connection Property Names & their Values (not including passwords).
Report_SQL	Memo	The SQL Query for the main report
Live_Data_1_0	Number	1 if user retrieved data from Database . 0 if saved data in the report was used.
Records_Read_N	Number	Number of Records Read into the Main_Report
Export_Activity_1_0	Number	1 (Yes) or 0 (False)
Print_Activity_1_0	Number	1 (Yes) or 0 (False)

SQL Server Data Structure

Here is a script for creating the table in Microsoft **SQL Server**:

```
CREATE TABLE [dbo].[DLV_Log] (
[LogN] [int] IDENTITY (1, 1) NOT FOR REPLICATION  NOT NULL ,
[Rpt_Path] [nvarchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Rpt_Name] [nvarchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Proc_Start_Local] [nvarchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Proc_End_Local] [nvarchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Proc_Start_GMT] [nvarchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Proc_End_GMT] [nvarchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Windows_User_ID] [nvarchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Machine_ID] [nvarchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Parameter_Values] [nvarchar] (40000) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Connection_Properties] [nvarchar] (40000) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Report_SQL] [nvarchar] (40000) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Live_Data_1_0] [int] NULL ,
[Records_Read_N] [int] NULL ,
[Export_Activity_1_0] [int] NULL ,
[Print_Activity_1_0] [int] NULL
) ON [PRIMARY]
GO
```

SQL Server notes:

- for recent versions of SQL Server, replace [nvarchar] (**40000**) with [nvarchar] (**MAX**)
- To ensure the User ID you use can access that table, make that user a member of the dbo_owner group in SQL Server.

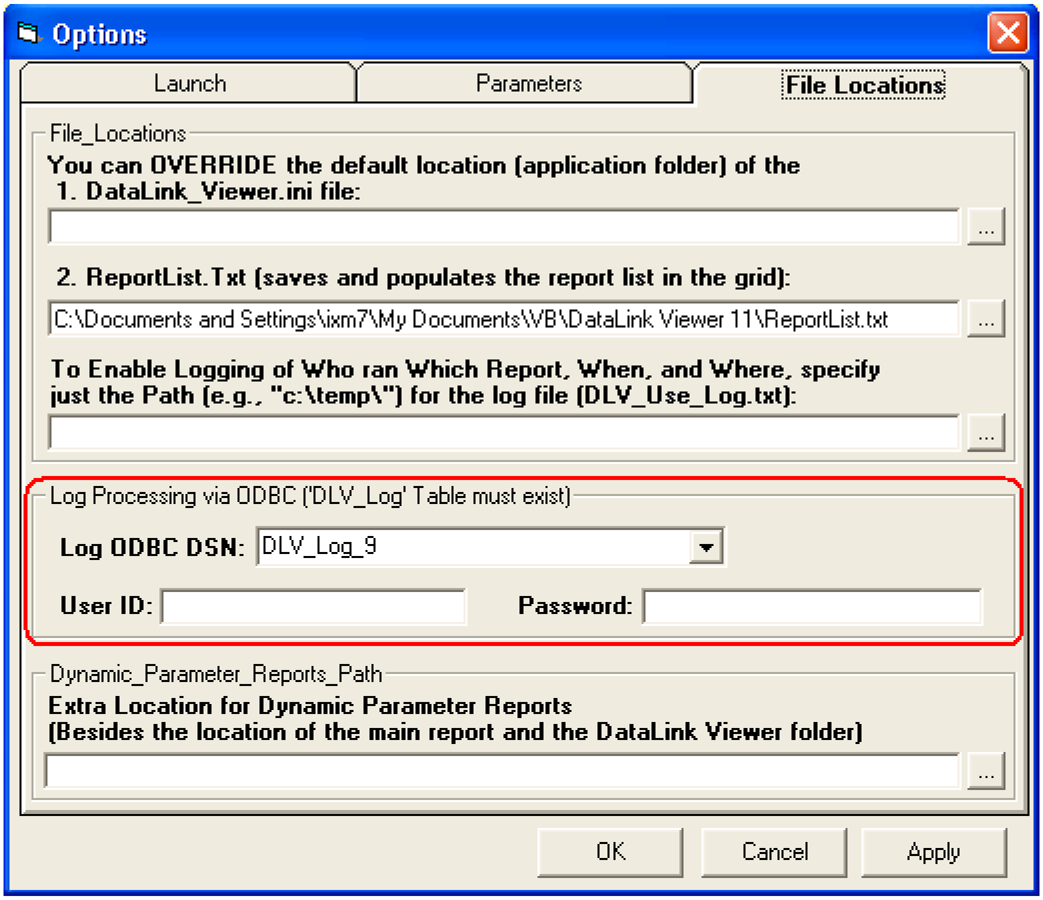
ORACLE DATA STRUCTURE

The script below, contributed by Jonathan Klobucnik from A.H. Belo, also includes a definition of a Sequence and a Trigger to handle the Auto-Numbering requirement:

```
CREATE TABLE DLV_LOG
(
  LOGN NUMBER(*, 0) NOT NULL , RPT_PATH NVARCHAR2(255) , RPT_NAME NVARCHAR2(255)
, PROC_START_LOCAL NVARCHAR2(255) , PROC_END_LOCAL NVARCHAR2(255)
, PROC_START_GMT NVARCHAR2(255) , PROC_END_GMT NVARCHAR2(255)
, WINDOWS_USER_ID NVARCHAR2(255) , MACHINE_ID NVARCHAR2(255) , PARAMETER_VALUES CLOB
, CONNECTION_PROPERTIES CLOB , REPORT_SQL CLOB , LIVE_DATA_1_0 NUMBER(*, 0)
, RECORDS_READ_N NUMBER(*, 0) , EXPORT_ACTIVITY_1_0 NUMBER(*, 0)
, PRINT_ACTIVITY_1_0 NUMBER(*, 0) , CONSTRAINT DLV_LOG_PK PRIMARY KEY
  (
    LOGN
  )
  ENABLE
)
LOGGING
PCTFREE 10
INITRANS 1
STORAGE
(
  BUFFER_POOL DEFAULT
)
LOB (PARAMETER_VALUES) STORE AS
(
  ENABLE STORAGE IN ROW
  CHUNK 8192
  RETENTION
  NOCACHE
  LOGGING
)
LOB (CONNECTION_PROPERTIES) STORE AS
(
  ENABLE STORAGE IN ROW
  CHUNK 8192
  RETENTION
  NOCACHE
  LOGGING
)
LOB (REPORT_SQL) STORE AS
(
  ENABLE STORAGE IN ROW
  CHUNK 8192
  RETENTION
  NOCACHE
  LOGGING
);
/
/*Sequence to allow the trigger to "auto number" the primary key*/
CREATE SEQUENCE DLV_LOG_SEQ;
/
/*Trigger to automatically generate the primary key values*/
CREATE OR REPLACE TRIGGER DLV_LOG_INS
BEFORE INSERT ON DLV_LOG
FOR EACH ROW
WHEN (new.LogN IS NULL)
BEGIN
  SELECT DLV_LOG_SEQ.NEXTVAL
  INTO   :new.LogN
  FROM   dual;
END;
```

How to Start Logging?

The following DataLink Viewer Options dialog allows you to specify the ODBC Data Source Name (DSN) where the DLV table resides and the User ID & Password (stored encrypted), if the data source requires a login.



File_Locations

You can OVERRIDE the default location (application folder) of the
1. DataLink_Viewer.ini file:

2. ReportList.Txt (saves and populates the report list in the grid):

C:\Documents and Settings\ixm7\My Documents\VB\DataLink Viewer 11\ReportList.txt

To Enable Logging of Who ran Which Report, When, and Where, specify
just the Path (e.g., "c:\temp\") for the log file (DLV_Use_Log.txt):

Log Processing via ODBC ('DLV_Log' Table must exist)

Log ODBC DSN: DLV_Log_9

User ID:

Password:

Dynamic_Parameter_Reports_Path

Extra Location for Dynamic Parameter Reports
(Besides the location of the main report and the DataLink Viewer folder)

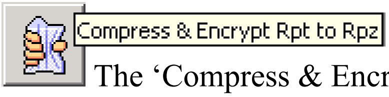
Record Report Use in a Text Log File

If you use the Options dialog to specify a folder location for a **DLV_Use_Log.txt** DLV creates that file (if it doesn't already exist) and appends to it use statistics after each report viewing. This log file contains one row with column headers followed by one row for each report viewing event. The columns are:

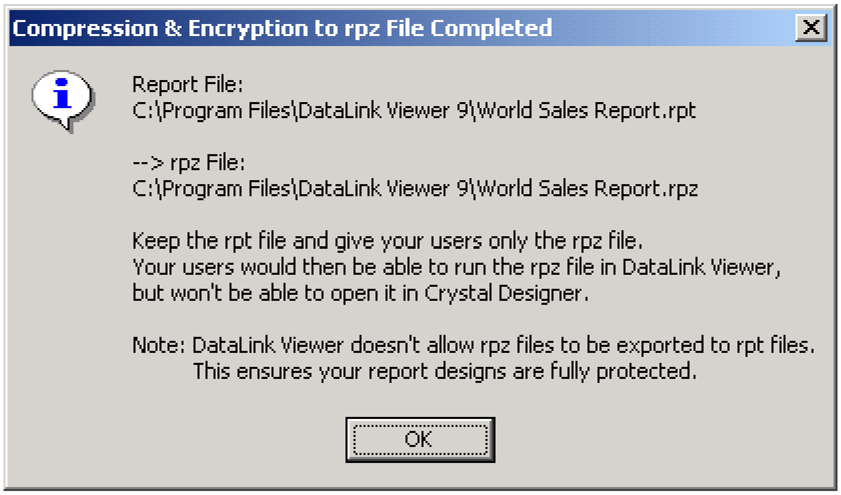
1) Report Path
2) Report Name
3) User_ID
4) PC
5) Start
6) Finish

You can use that file (for example, using Crystal with a Text ODBC driver) to analyze what reports are popular, who runs what reports, and what reports take too long to run.

Protect Report Designs with rpz Files



The 'Compress & Encrypt Rpt to Rpz button' (on the 1st Tab of DataLink Viewer) allows you to select an rpt file and convert it to **rpz** file.



Compression & Encryption to rpz File Completed

Report File:
C:\Program Files\DataLink Viewer 9\World Sales Report.rpt

--> rpz File:
C:\Program Files\DataLink Viewer 9\World Sales Report.rpz

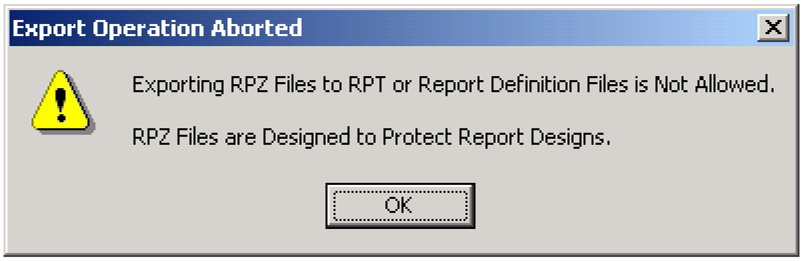
Keep the rpt file and give your users only the rpz file.
Your users would then be able to run the rpz file in DataLink Viewer,
but won't be able to open it in Crystal Designer.

Note: DataLink Viewer doesn't allow rpz files to be exported to rpt files.
This ensures your report designs are fully protected.

The resulting **rpz** file is a compressed and encrypted version of the rpt file that is recognizable only by DataLink Viewer & Visual CUT. Your users can run the resulting **rpz** files in DataLink Viewer or Visual CUT, but cannot view or modify them in Crystal.

This allows developers to protect and hide their reports designs (either as an intellectual property issue or as a tech support issue). They can simply keep the rpt files and distribute only the **rpz** files to their users. Note: .rpz files should not be renamed.

Within DataLink Viewer & Visual CUT, **rpz** files behave just like rpt files except that, in order to protect the report design, exporting **rpz** files to "rpt" format or to "report definition" format is blocked:



Export Operation Aborted

Exporting RPZ Files to RPT or Report Definition Files is Not Allowed.

RPZ Files are Designed to Protect Report Designs.

MAKE_RPZ COMMAND LINE ARGUMENT

If you need to automate the conversion of rpt files to rpz files using scheduled or automated processes, DataLink Viewer provides a command line API.
Your command line call should look like this (all in one line):

```
"C:\Program Files\DataLink Viewer 12\DataLink_Viewer_12.exe" "Make_RPZ:c:\test\*.rpt>c:\MyFolder>120"
```

After the Make_RPZ key word and the colon, the arguments are separated by “>” as a delimiter.
The three arguments are:

1. The path to the rpt file(s) to be converted.
   - you may specify wild cards (as demonstrated in the example above)
   - you may specify multiple paths separated by a semi-colon. For example:
   ```
   "Make_RPZ:c:\rptFolder1\*.rpt;c:\rptFolder2\*.rpt>c:\rpzFolder>120”
   ```

2. The target folder where the rpz files should be deposited

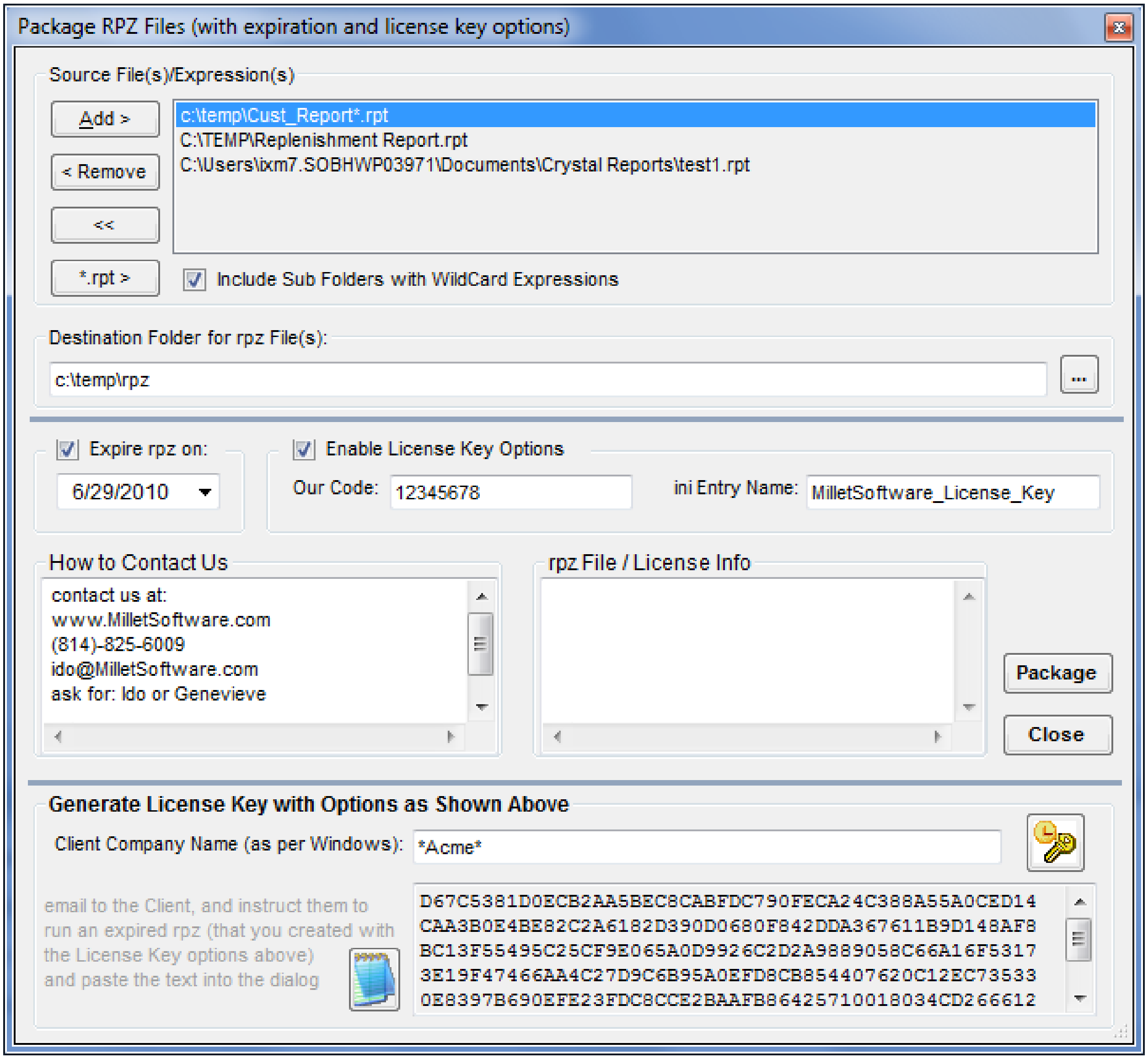
3. The maximum age, in minutes, of the rpt file in order to be included. Older files would be skipped. This allows the command line to be placed in a batch file and scheduled while avoiding the conversion of older files that have already been converted.
   Use a value of zero to convert all files regardless of age.

Create rpz Files with Expiration and License Keys

You can apply expiration date and/or license keys to rpz files by setting the following entry in the DataLink_Viewer.ini file:

```
    [Options]
    Enable_Advanced_RPZ_Options=True
```

When that option is enabled, the dialog you get when clicking on the 'Compress & Encrypt Rpt to Rpz' button looks like this:



Package RPZ Files (with expiration and license key options)

Source File(s)/Expression(s)

c:\temp\Cust_Report*.rpt
C:\TEMP\Replenishment Report.rpt
C:\Users\ixm7.SOBHWP03971\Documents\Crystal Reports\test1.rpt

☑ Include Sub Folders with WildCard Expressions

Destination Folder for rpz File(s): c:\temp\rpz

☑ Expire rpz on: 6/29/2010

☑ Enable License Key Options

Our Code: 12345678

ini Entry Name: MilletSoftware_License_Key

How to Contact Us

contact us at:
www.MilletSoftware.com
(814)-825-6009
ido@MilletSoftware.com
ask for: Ido or Genevieve

rpz File / License Info

Generate License Key with Options as Shown Above

Client Company Name (as per Windows): *Acme*

email to the Client, and instruct them to run an expired rpz (that you created with the License Key options above) and paste the text into the dialog

```
D67C5381D0ECB2AA5BEC8CABFDC790FECA24C388A55A0CED14
CAA3B0E4BE82C2A6182D390D0680F842DDA367611B9D148AF8
BC13F55495C25CF9E065A0D9926C2D2A9889058C66A16F5317
3E19F47466AA4C27D9C6B95A0EFD8CB854407620C12EC73533
0E8397B690EFE23FDC8CCE2BAAFB86425710018034CD266612
```

Please contact Millet Software for more detail about the various options and the use scenarios supported by this dialog.

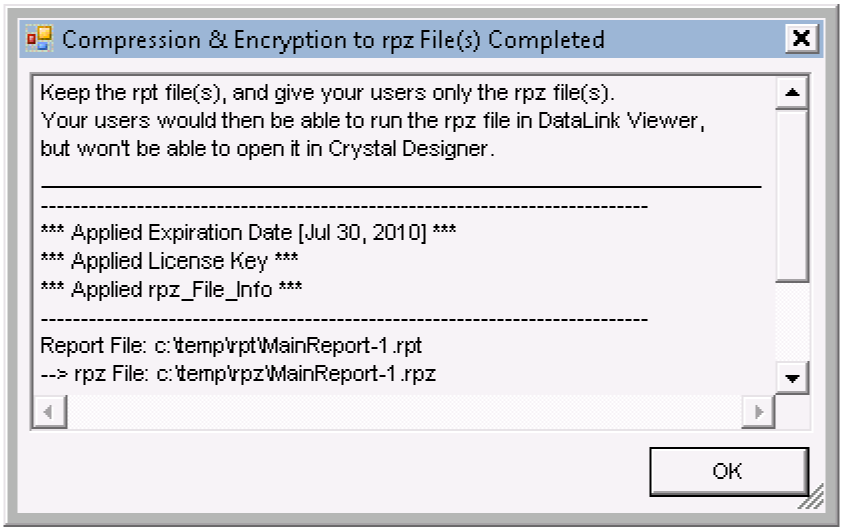
MAKE_RPZ2 COMMAND LINE ARGUMENT

If you need to automate the conversion of rpt files to rpz files with expiration dates and license keys, your command line call should look like this (all in one line):

```
"C:\Program Files\DataLink Viewer 11\DataLink_Viewer_11.exe" "Make_RPZ2:c:\test\*.rpt>>>
c:\MyFolder>>>False>>>2221440>>>rpz file info line 1/////and line 2>>>30/07/2010>>>
MilletSoftware_License_Key1>>>12345678>>>Call us at (814) 825-6008/////ido@MilletSoftware.com>>>
True"
```

After the Make_RPZ2 key word and the colon, the arguments are separated by “>>>” as a delimiter. The 8 arguments are:

1. The **path to the rpt file(s) to be converted**.
   - you may specify wild cards (as demonstrated in the example above)
   - you may specify multiple paths separated by a semi-colon.
2. The **target folder** where the rpz files should be deposited
3. **Include subfolders** when wildcard expressions are used for source files (True/False)?
4. The **maximum age**, in minutes, of the rpt file(s) to be included. Older files would be skipped. This allows the command line to be placed in a batch file and scheduled while avoiding the conversion of older files that have already been converted.
   Use a value of zero to convert all files regardless of age.
5. **Text information to embed in the rpz file**. Separate lines with ‘/////’. This information can be accessed via a report parameter named DLV_Rpz_File_Info
6. **Rpz Expiration Date** (dd/mm/yyyy). The rpz file will not run beyond that date, unless a license key with a later expiration date is added to the machine.
7. **Contact Information** to embed in the rpz file. This information is provided to the user when trying to run an expired rpz or an rpz requiring a new license key.
8. **Show Conversion Results Dialog** (True/False). This controls whether the process will be “quiet” (typical for a scheduled process) or result in a dialog showing what files were converted and what options were applied:



Compression & Encryption to rpz File(s) Completed

```
Keep the rpt file(s), and give your users only the rpz file(s).
Your users would then be able to run the rpz file in DataLink Viewer,
but won't be able to open it in Crystal Designer.

----------------------------------------------------------------------------------
*** Applied Expiration Date [Jul 30, 2010] ***
*** Applied License Key ***
*** Applied rpz_File_Info ***
----------------------------------------------------------------------------------
Report File: c:\temp\rpt\MainReport-1.rpt
--> rpz File: c:\temp\rpz\MainReport-1.rpz
```

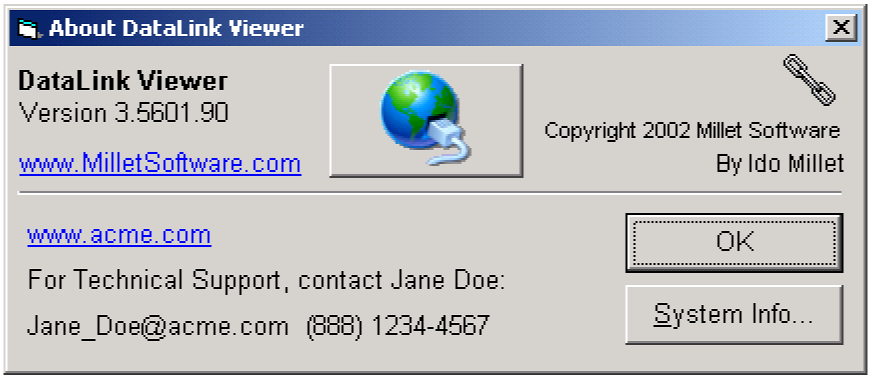
Add your Company Info to the About Dialog

In DataLink_Viewer.ini you can **set 3 lines in the About dialog to text of your choice.** This can be useful in a large company where you want users to know **who to contact with technical questions**. It can also be useful when you sell Crystal reports bundled with DataLink Viewer and you wish to specify your contact information for similar reasons.

For example, using the following settings in DataLink_Viewer.ini:

```
[Options]
About_Line1=www.acme.com
About_Line2=For Technical Support, contact Jane Doe:
About_Line3=Jane_Doe@acme.com  (888) 1234-4567
```

you would get the following About dialog:
Note: as demonstrated with the **www.acme.com** line, DataLink Viewer automatically assigns a web link to lines that contain only a url.



About DataLink Viewer

DataLink Viewer
Version 3.5601.90
www.MilletSoftware.com

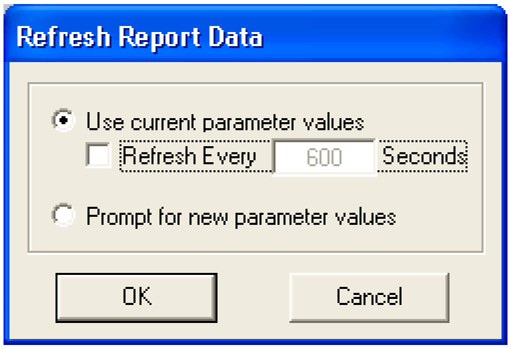
Copyright 2002 Millet Software
By Ido Millet

www.acme.com
For Technical Support, contact Jane Doe:
Jane_Doe@acme.com (888) 1234-4567

Auto-Refresh Reports Every N Seconds

When clicking the Refresh button, you can specify that the report should automatically refresh every N seconds.

As shown in the following dialog, when a report has parameters, the Auto-Refresh option is enabled only when you elect to reuse the current parameter values:



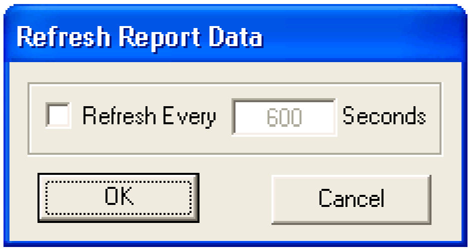
Refresh Report Data

● Use current parameter values

□ Refresh Every 600 Seconds

○ Prompt for new parameter values

When a report doesn't have parameters, the dialog is simpler and you can always elect to enable auto-refresh:



Refresh Report Data

□ Refresh Every 600 Seconds

The auto-refresh process stops when you click the Refresh button again or when you select another report.

Auto_Refresh Command Line Argument

You can also trigger auto-refresh by using an "Auto_Refresh" argument when invoking a report from a command line. Here's how you would trigger a viewing of a report and auto-refreshing it every 5 minutes (300 seconds):

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\Report.rpt"
"Auto_Refresh:300"
```

ViewMode Command Line Argument (Remove Toolbar/Status bar)

When you trigger a report from a command line using a –v (View Only window), you can also elect to remove the toolbar and/or the status bar in order to maximize the space available for the report itself.

The command line argument for controlling this behavior is "ViewMode" and it has 3 possible values:
…"ViewMode:**NoStatusBar**" - only the status bar at the bottom of the window is removed
…"ViewMode:**NoToolBar** - only the toolbar at the top of the window is removed
…"ViewMode:**NoBars**" - both the toolbar and the status bar are removed

A typical scenario for using this option is in combination with the Auto_Refresh command line argument. For example:

```
"C:\Program Files\DataLink Viewer 12\DataLink_Viewer_12.exe" -v "C:\temp\Report.rpt"
"Auto_Refresh:300" "ViewMode:NoBars"
```

Cycling Through Several Auto-Refreshed Reports

Assume you want your monitor to cycle through 2 different auto-refreshed reports every 15 seconds. You can take advantage of the fact that as a report gets auto-refreshed its window becomes the active one. So, you can trigger auto-refresh every 30 seconds for both reports, but separate the start times by 15 seconds.

Here’s how you can automate the whole process:

Step 1: Use notepad to create a batch file that can be called from within another batch file to delay processing by a given number of second. Call it **WAIT.bat**
(if you’d like to understand the logic, see: http://malektips.com/dos0017.html):

```
@ping 127.0.0.1 -n 2 -w 1000 > nul
@ping 127.0.0.1 -n %1% -w 1000> nul
```

Step 2: Use notepad to create the following text file. Call it **invis.vbs**
(this allows us to call 1 report and progress to the next line in a batch file (no blocking) as well as avoid the ugly DOS window):

```
CreateObject("Wscript.Shell").Run """" & WScript.Arguments(0) & """", 0, False
```

Step 3: Use notepad to create one batch file for each report to trigger its viewing and auto-refreshing. Here are two batch files, one for a report with a parameter and one for a report without one:

Report_1.bat

```
"C:\Program Files\DataLink Viewer 11\DataLink_Viewer_11.exe" -v "C:\DLV\Report_1.rpt" "Parm1:2004" "Auto_Refresh:30"
```

Report_2.bat

```
"C:\Program Files\DataLink Viewer 11\DataLink_Viewer_11.exe" -v "C:\DLV\Report_2.rpt" "Auto_Refresh:30"
```

Step 4: Use notepad to create a batch file, which silently calls the two reports, inserting a 15 seconds wait between the calls. Double-click or schedule that batch file to start the processing:

```
wscript.exe "C:\DLV\invis.vbs" "C:\DLV\Report_1.bat"
CALL WAIT 15
wscript.exe "C:\DLV\invis.vbs" "C:\DLV\Report_2.bat"
```

Click to Change a Parameter Value

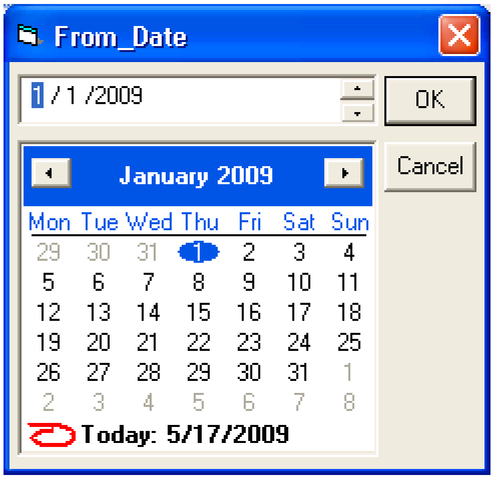
If you place a main report parameter on the report layout, when a DataLink Viewer user clicks on that parameter, DataLink Viewer prompts the user for just that parameter. That allows your users to simply and intuitively select just one parameter to refresh.

Toggling Between 2 Parameter Values

If a user clicks on a parameter that has **exactly 2 default values** (no custom values allowed), DataLink Viewer toggles the value of that parameter to the other value. This provides many useful behaviors. The sample report demonstrates toggling between viewing Profit or Revenue by simply clicking on the {?Metric} parameter. Another parameter allows toggling between expanding all top nodes or manually selecting which nodes to expand.

Special Date Parameter Dialog

If a user clicks on a single-value date parameter, DataLink Viewer presents an enhanced date entry/selection dialog. To quickly select a date, **double-click a date** on the calendar. Once the calendar is selected, **Page Up/Down** changes the month and **Ctrl+(Page Up/Down)** changes the year:



1 / 1 /2009

January 2009

Mon	Tue	Wed	Thu	Fri	Sat	Sun
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

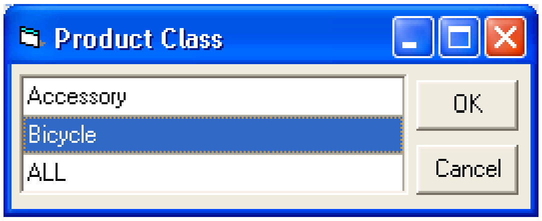
Today: 5/17/2009

Input Box Dialog

If a user clicks on a single-value parameter with **less than 2 default values**, where custom values are allowed, DataLink Viewer presents a simple InputBox dialog, making it easier and faster for the user to change the parameter value.

ListBox Dialog

If a user clicks on a single-value parameter **more than 2 default values**, where custom values are not allowed, DataLink Viewer presents a multi-row dialog, with one row for each default value designed for that parameter. This allows the user to quickly change the value for that parameter. To quickly select an item, the user can simply double-click it.



.

Changing a Parameter by Clicking on a Related Formula

In some cases, you may wish to invoke a parameter change by clicking on a related formula. For example, you may wish to embed the parameter inside some text, or perhaps the parameter allows multiple values (making simple value display impossible), or a parameter may have an empty value, making it impossible to click on the parameter object by itself.

To use a formula as a proxy for clicking on a parameter, the formula name must start with "**DLV_Param_Click:**" followed by the parameter name you wish to change when a user clicks on that formula. For example, "DLV_Param_Click:Customers"

The formula itself may display any values you wish. For example, you may concatenate static text with the parameter value, or you may wish to return "No Value" if the parameter value is null, and a nicely formatted value if the parameter is not null..

Click to Set Formula Value

DataLink Viewer reacts to user clicks on specially named formulas by setting the value of another formula. For example, if a user clicks on a formula called `{@DLVSet_1:Sort1}` that has as its string value the text "Region" it will attempt to locate the {@Sort1} formula and set its string value to "Region". If, the user clicks on another formula called `{@DLVSet_2:Sort1}` that has as its string value the text "Country" it will attempt to locate the {@Sort1} formula and set its string value to "Country".

The name of the clicked formula has 4 parts:
1. **DLVSet_** this is a constant that identifies the formula to DataLink Viewer as a click source.
2. Any number (not limited to a single digit) (1 or 2 in the examples above). This allows for multiple formulas, each with a different string value, to set the string value of the same target formula
3. a colon (**:**)
4. The name of the target formula (Sort1 in the examples above) whose string value would be set to the string value of the clicked formula.

With some creativity, this allows you to design reports that react to user clicks in useful ways.

Click Column Headers to Re-Sort The Report

As a demonstration of what can be achieved with the "Click to Set a Formula Value" functionality, the sample report (DLV_Column_Header_Sort) reacts to clicks on column headers by resorting the report based on the clicked column header. A short video demonstration is available at: http://www.milletsoftware.com/Download/DLV_Sort_By_Click.wmv

The four column headers are clickable formulas:

1. **{@DLVSet_1:Sort1}** (with a string value of "Customer Name")
2. **{@DLVSet_2:Sort1}** (with a string value of "Region")
3. **{@DLVSet_3:Sort1}** (with a string value of "Country")
4. **{@DLVSet_4:Sort1}** (with a string value of "Postal Code")

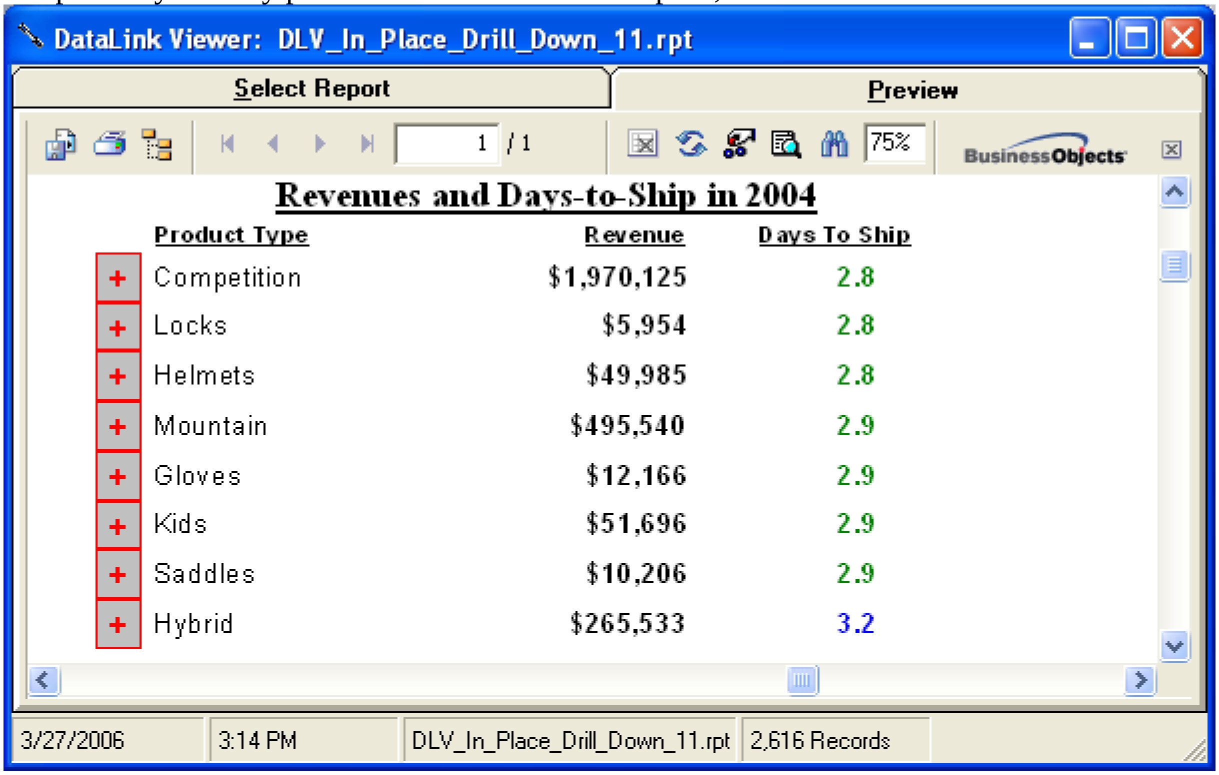
The target formula {@Sort1} starts with a design-time value of "Customer Name" and as its value gets changed due to user clicks, the {@SortBy1}formula reacts by returning one of the 4 database columns (Customer Name, Region, Country or Postal Code) . This in turn changes how the report groups are sorted.

Note: **group** rather than **record** sorting is used in this sample report because the Crystal runtime components don't re-sort records unless the report data is refreshed from the database. The Group re-sort occurs all on the client-side without hitting the database again.

In-Place Drill-Down

Crystal's Drill-Down functionality is limited to opening the detail in a new tab. However, in many cases users prefer to expand/collapse detail within the main report tab. By adding to the report a few formulas and using them in the suppress expression of detail sections, DataLink Viewer allows you to achieve In-Place Drill-Down functionality.

A report may initially preview with all details collapsed, like this:

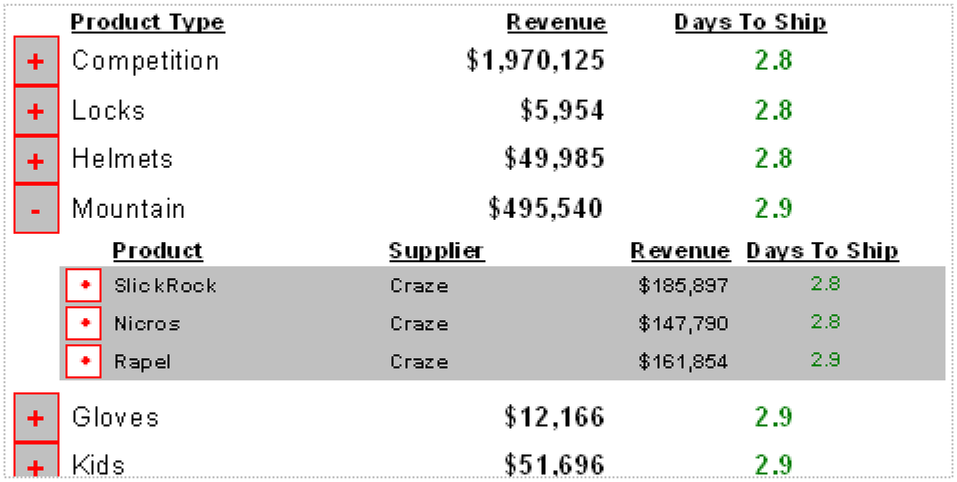


Revenues and Days-to-Ship in 2004

	Product Type	Revenue	Days To Ship
+	Competition	$1,970,125	2.8
+	Locks	$5,954	2.8
+	Helmets	$49,985	2.8
+	Mountain	$495,540	2.9
+	Gloves	$12,166	2.9
+	Kids	$51,696	2.9
+	Saddles	$10,206	2.9
+	Hybrid	$265,533	3.2

3/27/2006 | 3:14 PM | DLV_In_Place_Drill_Down_11.rpt | 2,616 Records

The + "buttons" are actually a formula that return a plus or a minus sign. When the user clicks on the these "buttons" the level-2 detail is revealed:

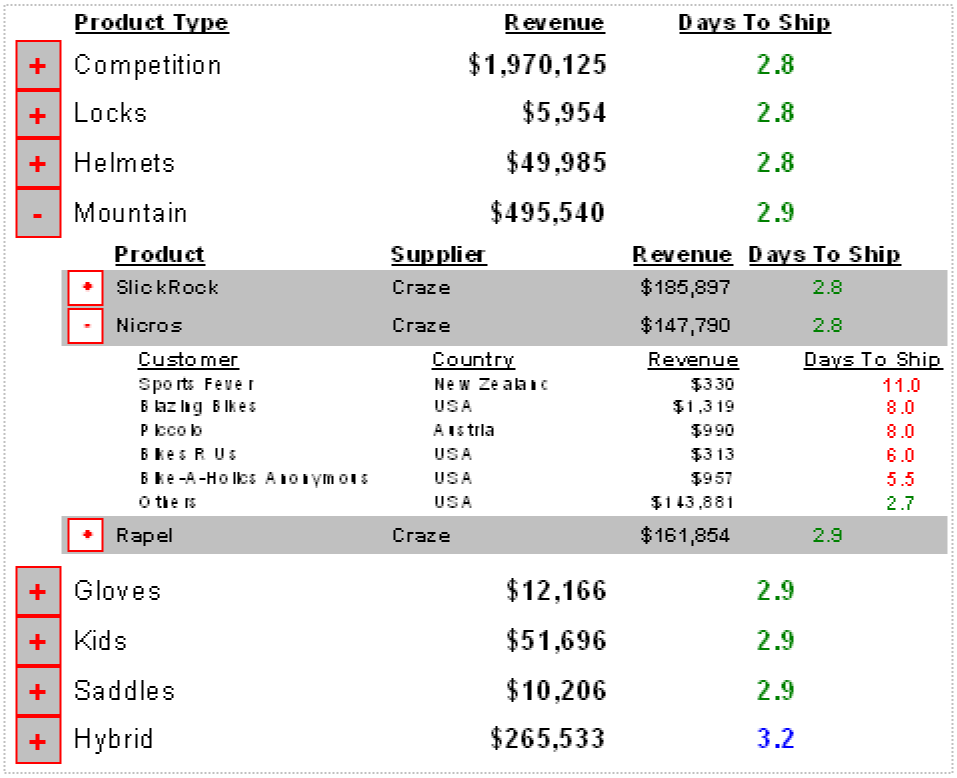


	Product Type	Revenue	Days To Ship
+	Competition	$1,970,125	2.8
+	Locks	$5,954	2.8
+	Helmets	$49,985	2.8
-	Mountain	$495,540	2.9

	Product	Supplier	Revenue	Days To Ship
•	SlickRock	Craze	$185,897	2.8
•	Nicros	Craze	$147,790	2.8
•	Rapel	Craze	$161,854	2.9

	Product Type	Revenue	Days To Ship
+	Gloves	$12,166	2.9
+	Kids	$51,696	2.9

And when a + "button" (another formula) in level-2 is clicked, level-3 detail is revealed:



	Product Type			Revenue	Days To Ship
+	Competition			$1,970,125	2.8
+	Locks			$5,954	2.8
+	Helmets			$49,985	2.8
-	Mountain			$495,540	2.9
		Product	Supplier	Revenue	Days To Ship
	+	SlickRock	Craze	$185,897	2.8
	-	Nicros	Craze	$147,790	2.8
		Customer	Country	Revenue	Days To Ship
		Sports Fever	New Zealand	$330	11.0
		Blazing Bikes	USA	$1,319	8.0
		Piccolo	Austria	$990	8.0
		Bikes R Us	USA	$313	6.0
		Bike-A-Holics Anonymous	USA	$957	5.5
		Others	USA	$143,881	2.7
	+	Rapel	Craze	$161,854	2.9
+	Gloves			$12,166	2.9
+	Kids			$51,696	2.9
+	Saddles			$10,206	2.9
+	Hybrid			$265,533	3.2

You are not limited to 1 drill-down path -- several branches can be expanded at the same time.

see video demo at: http://www.milletsoftware.com/Download/DLV_InPlace_DrillDown.wmv

The installation of DataLink Viewer includes a sample report (for example, DLV_In_Place_Drill_Down_9.rpt if you installed version 9). To learn how to create your own report with this functionality, open the sample report and look at the comments and expressions in:

5. **@DLV_Expand_Button_1** and **@DLV_Expand_Button_2** formulas
6. **@DLV_Current_Group_1** and **@DLV_Current_Group_2** formulas
7. **@DLV_Expanded_Group_List** formula
8. Suppress expressions for the sections:
    a. **GH1b** (acts as column headers for level 2) and **GH2a**
    b. **GH2b** (acts as column headers for level 3) and **GH3**

The same approach can be extended to any number of group levels.

This functionality is enabled only in the main report view (not in Crystal's drill-down tabs).

Note: using my CUT Light UFL, you can "persist" to an ini file the information in **@DLV_Expanded_Group_List** so that the report "remembers" its collapse/expanded state.

Converting Section Double-Click to In-Place Drill-Down

When you implement In-Place Drill-Down, as discussed above, you may wish to allow the user to trigger this behavior by double-clicking anywhere in the section (rather than just by clicking the ⊞ / ⊟ **@DLV_Expand_Button_N** formula placed in that section.

To override the default double-click behavior in Crystal (drill-down to a new tab) and divert the double-click action to an In-Place Drill-Down, simply use all Upper Case when naming the button formula: for example instead of
@DLV_Expand_Button_1
use
@DLV_EXPAND_BUTTON_1 .

Note: Verify that if you move the mouse cursor over the formula placed in the report section, the correct upper/lower case is shown. If you already created the formula and you wish to change it to upper case, you may need to propagate that change to the actual formula object in the report section by copying the formula (Ctrl-drag) and then deleting the old clone.

Note 2: if only the 2nd click on the button is recognized, you can correct that problem using the same procedure as above: copy the button formula (using Ctrl-drag) and then delete the old clone.

Hiding the ⊞ / ⊟ Buttons

If you use an upper case formula name for any of your **@DLV_Expand_Button_N** formulas, you can hide the formula by using the section background color as the font color. This would allow you to use double-clicks exclusively for In-Place Drill-Down actions (the user will not see the ⊞ / ⊟ Buttons).

Disabling Report Preview Buttons

In some scenarios you may wish to remove the Print, Refresh, Export, Select Expert, or Search Expert buttons from the preview window of reports. This is useful, for example, in cases where your application launches reports in DataLink Viewer via a command line and you wish to restrict some users from printing and/or exporting the report.

You can globally remove the Print and/or Export buttons from the Preview window by setting the following options in the **DataLink_Viewer.ini** file:

```
[Options]
Disable_Print_Button=TRUE
Disable_Export_Button=TRUE
Disable_Select_Button=TRUE
Disable_Search_Button=TRUE
Disable_Refresh_Button=TRUE
```

Alternatively, you can control these options via **command line arguments**. For example (all in one line):

```
"C:\Program Files\DataLink Viewer 9\DataLink_Viewer_9.exe" -v "C:\temp\MyReport.rpt.rpt"
"Parm1:1997" "Disable_Print_Button:TRUE" "Disable_Export_Button:TRUE"
```

Note: disabling the Search Expert button, doesn't remove the simple search button. It removes the more advanced search button that allows users to actually browse into data fields.

Note: see "***Enforcing Settings in a Master DataLink_Viewer.ini File***" for information about enforcing these settings from the "master" ini file to all "user" ini files.

Disabling DataLink Viewer Buttons

In some scenarios you may wish to disable certain user interface buttons so that, for example, users can't check for online updates, convert rpt files to rpz files, open the user manual, etc.

You can disable user interface buttons by setting the following options in the **DataLink_Viewer.ini** file:

```
[Options]
Disable_RPZ_Creation=TRUE
Disable_Check_for_Updates=TRUE
Disable_Options_Dialog=TRUE
Disable_Browse_Dialog=TRUE
Disable_User_Manual=TRUE
Disable_Version_Info=TRUE
```

Note: see "***Enforcing Settings in a Master DataLink_Viewer.ini File***" for information about enforcing these settings from the "master" ini file to all "user" ini files.

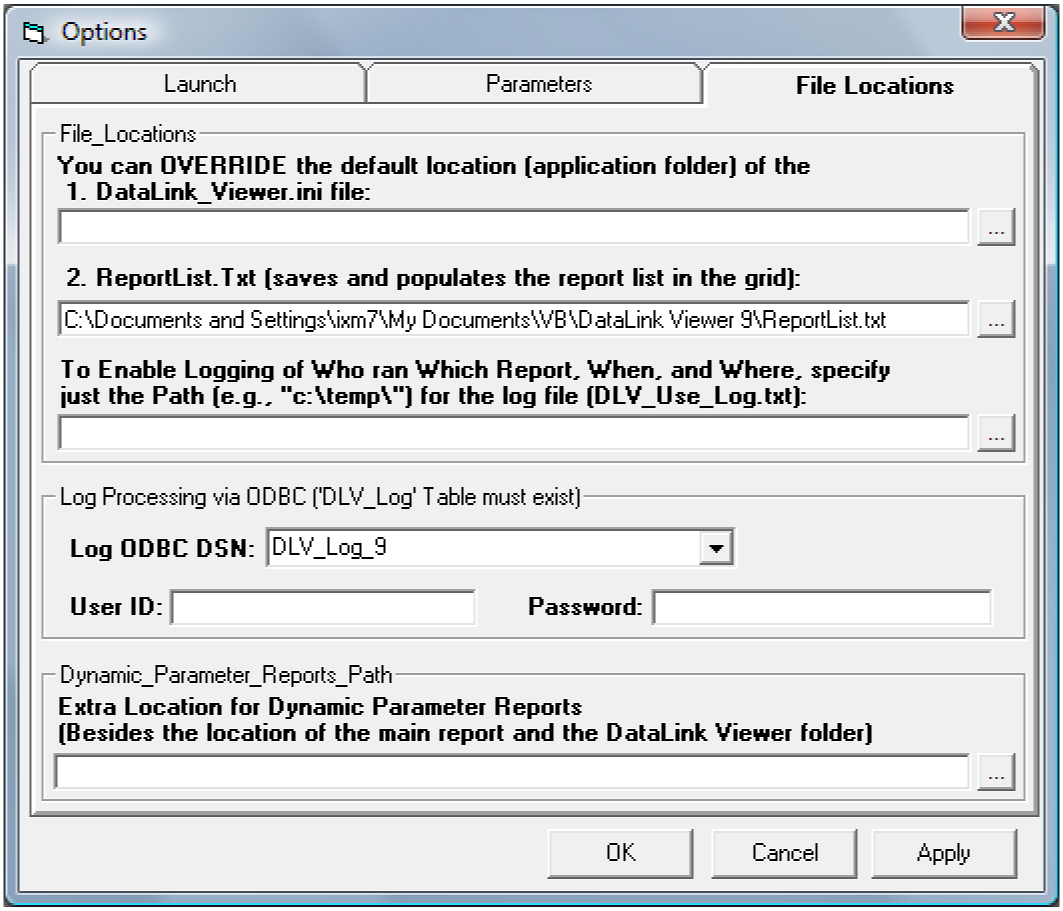
Citrix and File Location Functionality

DataLink Viewer can be installed on Citrix but be sure to install from the Console.

The **File Locations** tab in the Options dialog **allows a single installation of DataLink Viewer on a *Citrix* or *Terminal Server* to support individual settings for each user** by providing each user their own version of **DataLink Viewer.ini** and **ReportList.txt file** under their own mapped user drive. Note: upon initial launch of DataLink Viewer by a user, if the target **folders** don't exist, they get created, and if the target **files** don't exist, they get copied from the master copies in the application folder.

An alternative approach is to use these settings to force all users to share a centralized version of these files. In such a case, you should make the ReportList.txt folder read only to the users, so that the master list of reports doesn't get changed by the users.

This dialog also allows you to specify a centralized/local logging of report use statistics.



File_Locations

You can OVERRIDE the default location (application folder) of the
1. DataLink_Viewer.ini file:

2. ReportList.Txt (saves and populates the report list in the grid):
C:\Documents and Settings\ixm7\My Documents\VB\DataLink Viewer 9\ReportList.txt

To Enable Logging of Who ran Which Report, When, and Where, specify just the Path (e.g., "c:\temp\") for the log file (DLV_Use_Log.txt):

Log Processing via ODBC ('DLV_Log' Table must exist)

Log ODBC DSN: DLV_Log_9

User ID:

Password:

Dynamic_Parameter_Reports_Path

Extra Location for Dynamic Parameter Reports
(Besides the location of the main report and the DataLink Viewer folder)

EXPLICIT ASSIGNMENT OF DEFAULT PRINTER

To address a rare printing failure scenario under Citrix and Windows 2003, the following Datalink_Viewer.ini option allows an explicit assignment of the default printer when the user clicks the print button (before the printer selection dialog is displayed):

```
[Options]
Set_Report_To_Default_Printer=TRUE
```

SETTING INTEGRATED AUTHENTICATION FOR MULTIPLE MACHINES (CITRIX FARM)

The **Integrated_Authentication_Set_Entry** command line argument can automate encrypted storage (within DataLink_Viewer.ini) of login information (Database User ID & Password) per specified Windows User ID and Machine Name(s).

In the simplest case, this allows an administrator to set or change Integrated Authentication for a user after installation of DataLink Viewer or after database password changes. Keep in mind that this login information is stored encrypted and can be used only if the specified **Windows User ID** manages to log into the specified **Windows Machine Name**.

Automated setting of Integrated Authentication information can be particularly useful for Citrix or TS server farms. Such deployments allow each user to launch DataLink Viewer on the least busy machine within the server farm. Any changes made to the **DataLink_Viewer.ini** file in the user's *Home* folder are automatically replicated to the other machines participating in the server farm.

However, integrated authentication information is stored (encrypted) per user & **machine** combination. So, as an administrator, when you add a user to the server farm, you may need to generate Integrated Authentication entries for that user & **all machines in the server farm**. Multiple ini file entries for the same user & different machines look like this:

```
[Integrated_Authentication]
User33@Server101=148580D7008FB47042E07958BB8884A7BAD32B2B7C268A6...
User33@Server102=0AA0F62E3F8091CD90AD9D87F4F5279102439D2432134BF725…
```

To facilitate the automated generation of such entries, you can call DataLink Viewer using a command line (all in 1 line) argument as follows:

```
"C:\Program Files\DataLink Viewer 11\DataLink_Viewer_11.exe" "Integrated_Authentication_Set_Entry:
H:\DataLink_Viewer.ini>>Server101||Server102>>abc>>joe3>>sesame>>Append"
```

After the **Integrated_Authentication_Set_Entry** key word and the colon, the 6 arguments are separated by “>>” as a delimiter:

1. The **path & name of the ini file**
2. **List of machines separated by || delimiter (or just the name of 1 machine)**
3. **The Windows user id**
4. **The Database user id**
5. **The Database password**
6. **Operation Type:**
   a. **Append**: Appends the User ID & Password pair if it doesn't already exist for all specified machines
   b. **Replace**: like Append, but removes any other User ID & Passwords
   c. **Delete**: Deletes the User ID & Password pair if it already exists
   d. **Drop**: Completely drop all pairs from the specified User ID & Machine entries

Setting Encrypted Password Entries

The **Encrypted_Password_Set_Entry** command line argument can automate encrypted storage (within DataLink_Viewer.ini) of passwords. This allows an administrator to set or change encrypted passwords for DataLink Viewer or Visual CUT that can later be referenced from command line arguments.

To generate such entries in a targeted ini file, you call DataLink Viewer using a command line (all in 1 line) argument as follows:

```
"C:\Program Files\DataLink Viewer 11\DataLink_Viewer_11.exe" "Encrypted_Password_Set_Entry:
H:\DataLink_Viewer.ini>>Options>>Encrypted_Password_FTP>>sesame"
```

After the **Encrypted_Password_Set_Entry** key word and the colon, the 4 arguments are separated by “>>” as a delimiter:

1. The **path & name of the ini file**
2. The **ini section name** (typically '**Options**')
3. **The Password Name**
4. **The Password to be Encrypted**

File Location Redirect Logic for DataLink_Viewer.ini

Each time DataLink Viewer loads, if the user doesn't have write permissions on the DataLink_Viewer.ini file in the application folder (typical for Vista & Windows 7 machines), the location of the Master DataLink_Viewer.ini is redirected by following these steps:

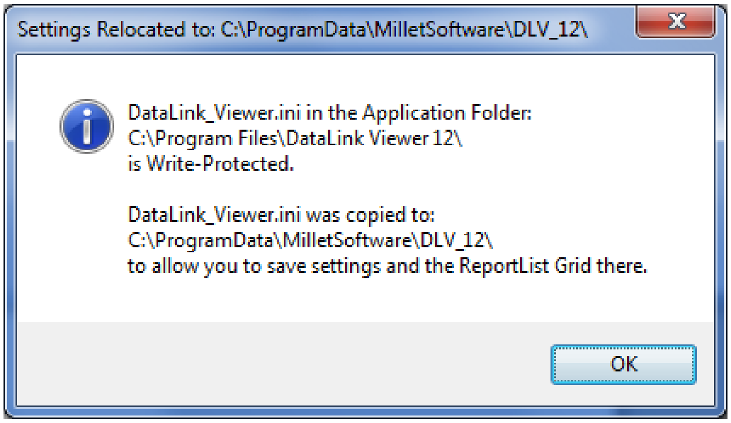
1. If the ini file in the application folder has a value specified for "ini_file" in the [File_Locations] section, that location is used for the DataLink_Viewer.ini
   note: this allows an administrator who has write permissions on the application folder to manually set the location.

2. More typically, the value in step 1 would be found as blank. DataLink Viewer then determines if a redirect is needed by testing to see if: a) ReportList.grd doesn't exists in the application folder **or** b) user doesn't have write permissions to DataLink_Viewer.ini in the application folder. If a redirect is needed, the application tries to locate and use DataLink_Viewer.ini in one of the following app data folder locations,
   under a **\MilletSoftware\DLV_N\** folder (**N** = **12**, **11**, **09**, or **08** depending on version):
   a. **Common App Data folder** - that location allows all users on that local machine to share the same ini file

   b. **User's Local App Data folder -** that location allows each user on that machine to have their own ini file

   c. **User's Roaming App Data folder -** that location allows each user to have the same ini file "follow them" to other machines.

3. If DataLink_Viewer.ini was not found in any of the 3 locations above, which is **typical in 1st-time installation scenarios**, DataLink Viewer will copy it as well as ReportList.txt from the application folder to the **Common App Data folder** and notify the user that the redirect has occurred. The location of ReportList.txt would also be set to the same redirected location. Here is what the notification looks like:.



Settings Relocated to: C:\ProgramData\MilletSoftware\DLV_12\

DataLink_Viewer.ini in the Application Folder:
C:\Program Files\DataLink Viewer 12\
is Write-Protected.

DataLink_Viewer.ini was copied to:
C:\ProgramData\MilletSoftware\DLV_12\
to allow you to save settings and the ReportList Grid there.

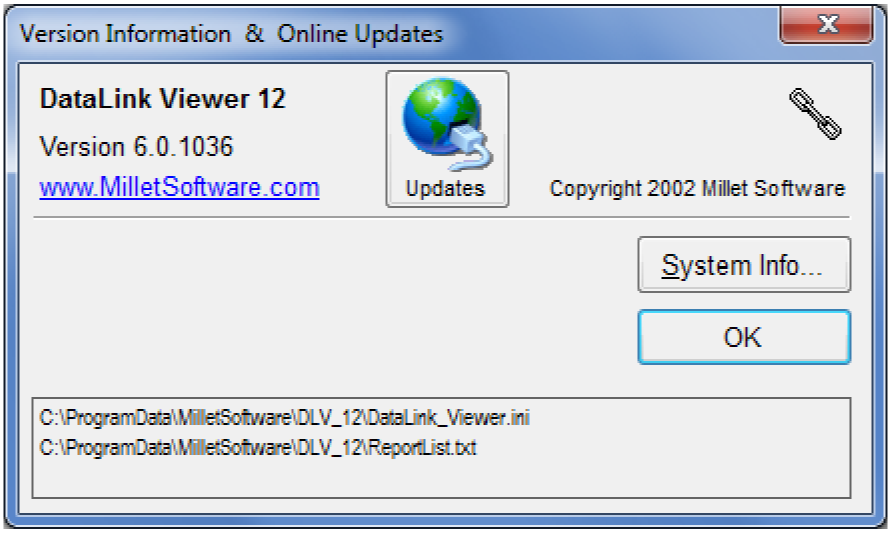
Continued on next page…

Notes: an ini file that was redirected to one of the locations above due to a write-protected application folder is considered a Master ini file. As usual, that Master ini file can be redirected further to a slave ini file by manually setting its “ini_file” location option under [File_Locations].

A blank option for ReportList.txt location in a redirected ini file is considered as an indication that the ReportList.txt file should be automatically copied to (if missing) and used at that location.

Checking and Navigating to Key File Locations

To check the locations of DataLink_Viewer.ini and ReportList.txt, you can click on the **‘Version Information & Updates’ button**. The textbox at the bottom of the dialog shows the current file locations:



```
C:\ProgramData\MilletSoftware\DLV_12\DataLink_Viewer.ini
C:\ProgramData\MilletSoftware\DLV_12\ReportList.txt
```

To navigate to the Folder where DataLink_Viewer.ini is located, **double-click that textbox** and DataLink Viewer will open that folder location in File Explorer. That makes it easier to find and edit the ini file in cases where the folder is hidden (typical for app data folders).

Enforcing Settings in a Master DataLink_Viewer.ini File

The “master” DataLink_Viewer.ini file is located in the application folder. As described above, the [File_Locations] section in the “master” file may indicate that a different “user” DataLink_Viewer.ini file, located at a different folder, should be used. As an administrator, you may want to ensure that some centralized settings in the “master” ini file always override the settings in the “use” ini file. You can do this by setting the values to all upper case (for example, TRUE or FALSE instead of True or False). This tells DataLink Viewer that it should use these settings from the “master” file.

This behavior applies to the following options, shown as if you wanted to apply them centrally from the “master” DataLink_Viewer.ini file:

```
[Options]
Disable_RPZ_Creation=TRUE
Disable_Check_for_Updates=TRUE
Disable_Options_Dialog=TRUE
Disable_Browse_Dialog=TRUE
Disable_User_Manual=TRUE
Disable_Version_Info=TRUE

Disable_Print_Button=TRUE
Disable_Export_Button=TRUE
Disable_Select_Button=TRUE
Disable_Search_Button=TRUE
```

Updating DataLink_Viewer.ini via a Delta File

When deploying DataLink Viewer to many users, you may want to automate the process of updating some of the DataLink_Viewer.ini settings. You can do that by placing a **DataLink_Viewer_Delta.ini** file in the application folder. Any entries found in that file will update DataLink_Viewer.ini when the application is launched.

Here is an example of a **DataLink_Viewer_Delta.ini** file:

```
[Delta_Options]
Delete_After=NEVER
//USE or NEVER or Some Date specified as yyyyMMdd
// if entry above not found, then default is USE
Update_Master_INI=False
// if entry above not found, then default  is False
Update_Slave_INI=True
// if entry above not found, then default is True

[Options]
Attempt_Logon_Without_Password=False
Strip_Table_Qualifiers=True
Saved_Data_Action=Display
Associated_File_Launch_Mode=View Only
New_Window_On_DblClick=True
Parameter_Values_Remember_Max_Chars=900
Saved_Parameter_Set_Minimum_N=5

[Integrated_Authentication]
Enable_Integrated_Authentication=True
```

The [Delta_Options] section is used only to control the following aspects of the process:

- The Delete_After option controls when the Delta ini file is deleted:
    - NEVER - the file will not be deleted
    - USE - the file is deleted after being used once
    - yyyyMMdd - the file is deleted after the specified date

- Update_Master_INI controls whether the Master ini file is updated.

- Update_Slave_INI controls whether the User ini file is updated.

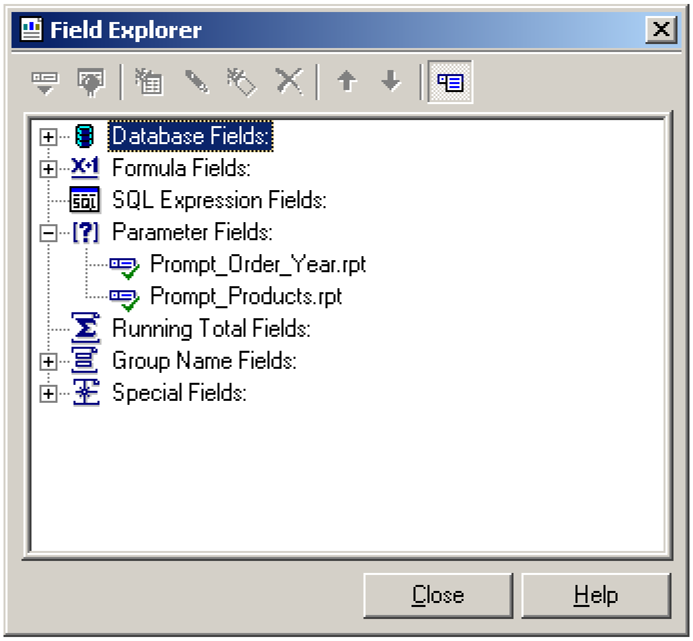
Dynamic and Cascading Parameters (Developer Notes)

DataLink Viewer can be used to run existing Crystal reports without any design changes.

This section reviews how report designers can activate the extra functionality of linking to live prompt reports.

Naming Parameters

If you want to link a parameter to a live prompt report, you need to name the parameter as the report name. In our sample report, the live prompt reports and parameter names were: "Prompt_Order_Year.rpt" and "Prompt_Products.rpt":



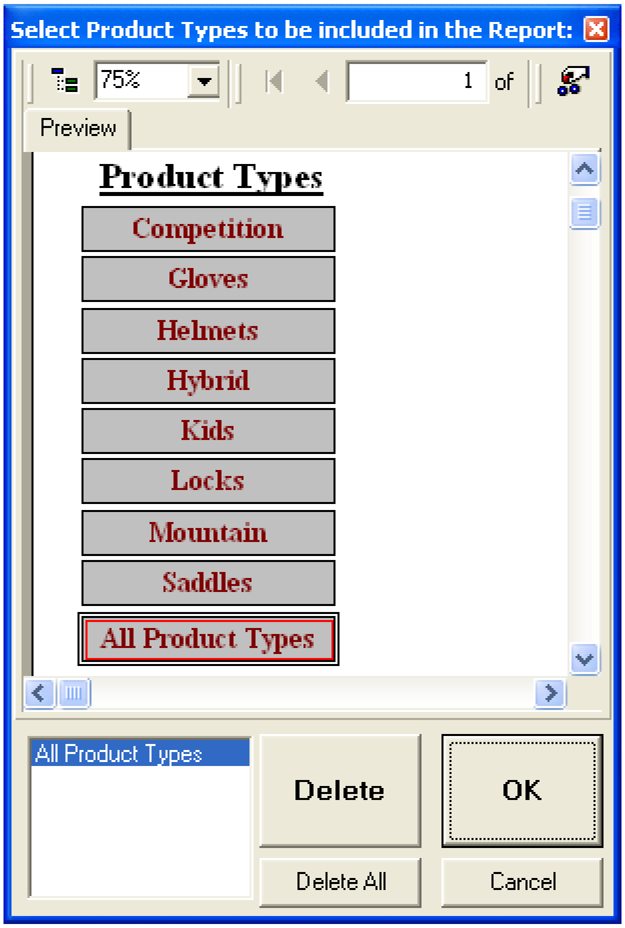
Designing Live Prompt Reports

The reports acting as live prompts should be **placed in the same directory as the report to be viewed or in the DataLink_Viewer directory**.
These reports **should contain at least one formula with the word "parameter" within its name**. Formulas named in such a manner can be clicked and selected within the prompt viewer.

You can place one formula on the report to let the user select a meaningful **Display Value** (e.g., "@Product_Name_Parameter") and place on the same section a "@Parameter_Data" formula to supply the actual **Data Value** passed into the parameter. You can't suppress the "@Parameter_Data" formula, but you can hide it by selecting a font color (e.g., White) and/or moving it to far right margin of the report. Note: Open the sample **Prompt_Products.rpt** report to see how the above is implemented.

Implementing a “Select All” Option

If you run the report again and scroll to the bottom of the live Product Type selection prompt, you would notice that you can select the “All Product Types” option:



This was implemented by adding a formula to the report footer with a static text value of “All Product Types”. The desired effect is obtained by specifying the following record selection criterion in **Prompt_Products.rpt**:

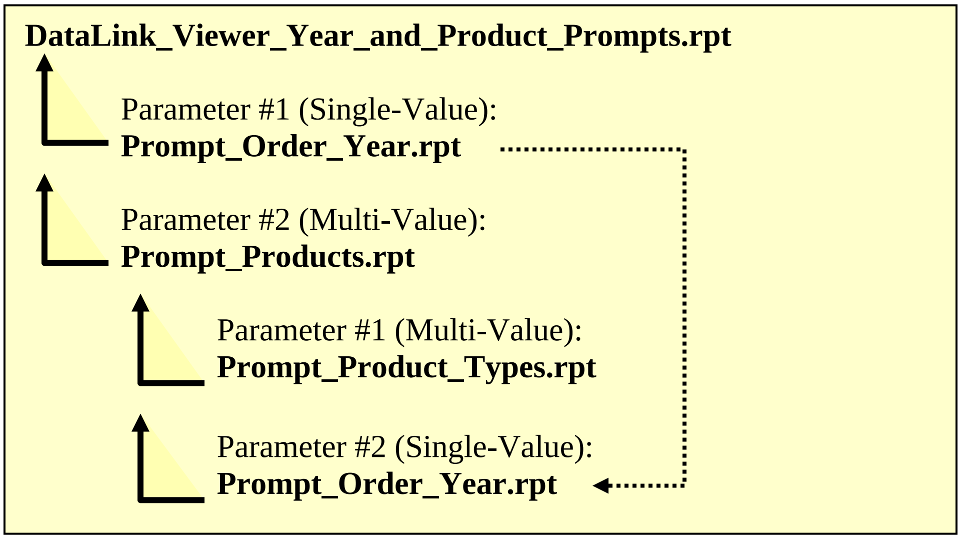
```
{Product_Type.Product Type Name} in {?Prompt_Product_Types.rpt}
OR
"All Product Types" in {?Prompt_Product_Types.rpt}
```

Sharing Values Across Cascading Parameters

Parameter values are automatically reused if the same Parameter name is used more than once in the chain of cascading parameters.

The diagram below shows a situation where **Prompt_Order_Year.rpt** is used to first ask the user to specify which orders should be shown in the report. That same parameter is then also used to limit the dynamic list of products (**Prompt_Products.rpt**) such that only products that actually sold in that year are available for selection.

DataLink Viewer now reuses the year specified by the user the first time **Prompt_Order_Year.rpt** is invoked to avoid asking the user the same question twice.



DataLink_Viewer_Year_and_Product_Prompts.rpt

Parameter #1 (Single-Value):
Prompt_Order_Year.rpt

Parameter #2 (Multi-Value):
Prompt_Products.rpt

Parameter #1 (Multi-Value):
Prompt_Product_Types.rpt

Parameter #2 (Single-Value):
Prompt_Order_Year.rpt

Note: A **DataLink_Viewer.INI** file is automatically created the first time *DataLink Viewer* links to a live prompt report. To disable sharing parameter values from prior answers, open this ini file using any text editor and change the following entry from TRUE to FALSE:

```
[Options]
DataLink_Parameters_Share_Prior_Answers=TRUE
```

Controlling the Default Value in Dynamic Parameters

As explained earlier in this manual, if you refresh or reload the same report, the values you specified last time for each dynamic parameter behave as defaults (they are already selected), so you can accept (Hit Return or click OK) or replace them. You can turn this feature off by un-checking "Remember Dynamic Parameter Prior Value' in the Options dialog.

There are cases where you may want to use logic within the dynamic parameter report to control what is shown as the default (already selected) value. For example, you may want to use the last order in the dynamic list as the default.

This can be achieved by following these guidelines:

1. add a string formula with the name of: **{@DLV_PARAMETER_DEFAULT_VALUE}** to the Report Footer (of the dynamic parameter report, not the main report).

    a. This formula can provide a **simple value**. For example, if you use:
    `Cstr({Order_N},0,"")`
    Since the formula is placed in the report footer, it will return the order number of the last order in the list.

    b. Alternatively, the formula can populate both the **display value as well as the data value** by separating them with `"||"` For example, if you use:
    `{Customer_Name} & "||" & {Customer_ID}`
    The default value the dialog would show would be the name of the last customer, but the actual parameter value (shown as a pop-up when you move your cursor over the selected value), would be the customer id.

    c. If the parameter is a multi-valued parameter, separate the values (formed as either a or b above) with `"&&"`. For example:
    `Descent&&Endorphin&&Triumph Pro Helmet`
    or
    `Descent||101&&Endorphin||103&&Triumph Pro Helmet||220`

2. The formula, and even the report footer section as a whole can be suppressed.

3. You must place the "Page N of M" special field in the Page footer of the dynamic parameter report (even if that field or the page footer as a whole are suppressed.

Requiring a Value for a Dynamic Parameter

This can be achieved by following these guidelines:

1. Add a **string** formula with the name of: **{@DLV_PARAMETER_REQUIRED}** to the Report Footer (of the dynamic parameter report, not the main report).

2. The formula must **return a non-blank string** if you wish to enforce a required value for the parameter.

3. If the string returned by the formula is longer than 10 characters it would be used as the text of the message box in cases where the user clicks OK without specifying a value for the parameter. Otherwise, a standard message appears.

4. The formula, and even the report footer section as a whole can be suppressed.

5. You must place the "Page N of M" special field in the Page footer of the dynamic parameter report (even if that field or the page footer as a whole are suppressed.

Forcing Users to Select from the List of Values (No Direct Edit)

By default, dynamic parameter dialogs for single-value parameters allow the user to enter a value directly instead of clicking on the displayed list. **You can force the user to select from the list displayed by the dynamic parameter report** under two scenarios:

1. You include the text '**NoEdit**' in the parameter name.

- or-

2. The user double-clicked an item in the dynamic parameter report and that item has an associated data value (a formula named Parameter_Data). If the option to remember dynamic parameter values is turned on, from that point on that parameter will force the user to select from the list.

Update History

Version 6.5002 Released August 29, 2016

- Packaged with a newer chilkatax-9.5.0-win32.dll component (version 9.5.0.58) to avoid dll conflict with the newer Visual CUT and CUT Light versions.

Version 6.5001 Released November 25, 2015

- Using DataLink Viewer XI R2, and the "Print_Copies:" argument. you can now set custom text on each print copy (such as 'Copy 1 of 2', 'Copy 2 of 2').
  See 'Setting Custom Text for Each Print Copy' and this demo image.

- Packaged with a newer chilkatax-9.5.0-win32.dll component (version 9.5.0.49) to avoid dll conflict with the newer Visual CUT and CUT Light versions.

- When starting DataLink Viewer with a command line argument of "ODBC_DSN:.." but no report path or other arguments, the ODBC DSN argument now applies to all reports opened during that session (rather than just to the first report opened).
  This allows a desktop shortcut, for example, to launch DataLink Viewer with a DSN target specified for all reports opened during that session.

Version 6.4001.0: Released November 27, 2014

- The Launch tab in the Options dialog now has a button called **'Encrypt & Save Password'**. It allows you to **centralize & protect passwords by avoiding specifying them directly inside command line argument**. Instead, you can name, encrypt and store the passwords inside DataLink_Viewer.ini. For detail, see "Referring to Saved Encrypted Passwords."

- In forced login scenarios, when integrated authentication is enabled, User ID changes now attempt to set a matching password even if the password entry is already populated.

- A new **Encrypted_Password_Set_Entry** command line argument allows administrators to **automate the saving of Encrypted Passwords to targeted ini files**. For detail, see "Setting Encrypted Password Entries."

- Fixed a problem with export to XML via command lines.

Version 6.3001.0: Released September 21, 2013

- **You can now schedule or trigger report exports (not just printouts) without user intervention.**

- Added support for loading rpz files packaged by much older versions of DataLink Viewer.

- Online Version Update functionality was removed due to the component Update.exe being (mistakenly) flagged by virus protection software.

- Fixed an issue with reading very large ini file entries (saved parameter values).

- When the login dialog shows alternative ODBC DSNs to select from, if the machine has more than 9 DSNs, the dialog is expanded and allows you to enter text to search the list of DSN. This makes it more efficient to locate a particular DSN in a long list of DSNs.

- Added an ini file option allowing users to disable setting initial directory when browsing to open rpt files: **OpenFileDialog_Set_Initial_Directory**=FALSE
  This solved a problem for one user in a Citrix/TS environment.

- You may now specify **ODBC_DSN_From_To** as a global entry in the [Options] section of DataLink_Viewer.ini. In cases where a global transition from old DSN to new DSN are desired, this removes the need to specify ODBC_DSN_From_To as a command line argument. Also, the entry as well as the command line argument **can now specify multiple pairs** separated by "||".

- **Database_Path_Selection** can now be specified via a command line argument.

- **Database_Path_Selection** can now be used for Paradox data sources by specifying just the path (not including the DB file name. For example, using a command line argument such as: `"Database_Path_Selection:C:\Old\>>c:\New\"`
  Note: this functionality is available only in DLV 11 and 2011.

- **A new Set_Formulas1 command line argument allows setting of formula expressions** (provided the formula name starts with a ^ character). For detail, see the user manual section on *Argument for Setting Formula Expressions*.
  Note: this functionality is available only in DLV 11 and 2011.

- **During forced login scenarios, Integrated Authentication now automatically sets the Password in the login dialog after the User ID is manually set.**

- A new **Integrated_Authentication_Set_Entry** command line argument allows administrators to **set integrated authentication information for specified Windows User ID and Machine Name(s)**. For detail, see "Setting Integrated Authentication for

Multiple Machines (Citrix Farm).”

- In the report grid, disabled delete action for a group to prevent accidental removal of all report rows belonging to that group.

- When closing the application after an interactive session, the ReportList.txt file (1st tab report grid) is now transferred to the recycle bin approximately once per 10 times. This provides a copy of the file in case it gets erased during abnormal termination of a user session.

- **Ctrl-Shift-F1** now opens DataLink_Viewer.ini (configuration settings) in Notepad.

- Folder selection functionality for Visual FoxPro (free tables) data sources that use a File DSN now applies to dynamic parameter scenarios on 64-bit as well as 32-bit machines.

- Depending on screen size, the selective parameter refresh grid now shows more rows (reducing the need to scroll when a report has many parameters).

Version 6.2001.0: Released May 30, 2011

- **DataLink Viewer now remembers parameter values and allows users to elect to reuse some or all of them**. This matches typical use scenarios and removes the need to enter all parameter values each time a user runs a report. The Options dialog (Process tab) allows users to control the maximum number of characters (default = 300) in parameter values that are saved for reuse or turn off completely the option to save and reuse parameters. The parameter values for each report are stored in DataLink_Viewer.ini. A 1-minute **video demo** is available.

- For situations where reports with many parameters are used in one of several parameter patterns, **you can now save & reuse Named Parameter Sets**. For detail, see "Save and Reuse Named Parameter Sets" or watch a **video demo**.

- In the Select Reports Tab, you can now use **Ctrl-F** to search for a row in the report list containing specified text. **F3** searches for the next matching row, and **Shift-F3** searches for a previous matching row.

- On initial install (or if the user can't update the DataLink_Viewer.ini in the application folder) DataLink Viewer creates an application settings folder, such as:
  \MilletSoftware\DLV_11\
  in the **common application settings directory** (e.g., c:\ProgramData) and redirects the locations of DataLink_Viewer.ini, ReportList.txt, and ReportList.grd to that folder. The user is notified when the initial redirect occurs and is advised to give the 'Users' group 'Modify' permissions on the new folder.
  Note: a blank option for ReportList.txt location in a redirected ini file is considered as an indication that the ReportList.txt file should be automatically copied to (if missing) and used at that location.

- The version information window now provides a text box with the file paths to the DataLink_Viewer.ini and ReportList.txt files. A double click on that text box opens that folder in File Explorer. The dialog also shows the 'Build Date' of the application.

- **You can now automate the process of updating entries in DataLink_Viewer.ini by placing an update file in the application folder**. For detail, see "Updating DataLink_Viewer.ini via a Delta File."

- **Added functionality for packaging rpz files with an expiration date and/or license key requirement**. This is useful for report developers who wish to provide customers with sample reports for an evaluation period or restrict report use to users who have a current subscription. Expiration and Licensing requirements are also enforced by Visual CUT. For detail, see "Create rpz Files with Expiration and License Keys."

- Fixed an issue with the report list grid not remembering expanded/collapsed state of some report groups. The problem existed only for versions before DataLink Viewer 2008 when the grid had many collapsed groups.

- Fixed an issue related to remembered window size and position when screen resolution has changed.

- About_Line options (in the DataLink_Viewer.ini file) now recognize all domains as live web links.

- Added a **Disable_Refresh_Button** option as an ini file entry or command line argument. This allows application developers to trigger viewing of a report in DataLink Viewer using specified parameters (via command line arguments) while ensuring the user cannot elect to run the report with different parameters.

- Fixed an issue causing ODBC DSN choices to be visible in the login window even when the report was not using ODBC as a connection method.

- The report loading now aborts (after an appropriate message) when command line arguments (ODBC_DSN or ODBC_DSN_From_To) specify DSNs that don't exist on the machine. The loading also aborts if an ODBC_DSN_From_To is specified as ODBC_DSN_**FROM**_To ('FROM' in all upper case) and none of the tables in the report or Subreports uses a DSN that matches the FROM argument. This allows application developers to ensure a report preview request doesn't return unexpected results.

- DataLink Viewer no longer issues a warning about a missing ODBC DSN setup when the report uses a File DSN.

- Added an **option to avoid attempts to connect to the database without a password**. By default, DLV first attempt to connect to the database without a password. If all your tables require a password, change the following option in the DataLink_Viewer.ini file from the default of TRUE to FALSE.
  This would reduce potential lockouts and failed login entries in the DBMS log.
  ```
  [Options]
  Attempt_Logon_Without_Password=FALSE
  ```

- Removed a component handling warnings from Microsoft Outlook during exports. This **removes the need for administrative rights to register the component**. DataLink_Viewer_Helper.exe is now used instead.

- Added an option to force the initial report browsing folder to a specific path (for example, C:\QA). You can set that option by starting DataLink Viewer with a command line argument like this:
  ```
  "c:\Program Files\DataLink Viewer 11\DataLink_Viewer_11.exe" "rpt_Folder:c:\QA"
  ```

- When clicking the User Manual button, the user now gets a message about the need to install a pdf file reader if the machine doesn't have an application associated with the .pdf file extension.

- When a user clicks on a parameter (on the report canvas) and changes its value, a **Loading, Please Wait** animation appears if the report refresh process takes a significant amount of time. Also, fixed a problem when the user then elects to load another report. The 2008 version of DLV did not have this problem.

- DataLink Viewer XI R2 now uses a newer version (Service Pack 6.4) of the Crystal XI R2 runtime components. Users of prior versions can't use the online update option and must go through a Remove & Install process in order to update their version.

Version 6.0001.0: Released January 19, 2010

- **DataLink Viewer now support .rpz files as dynamic parameter dialogs** (previously, only main report files were supported as .rpz files).

- **If a user clicks on a parameter (placed on the report layout), that has only 2 default values (with no custom values allowed), DataLink Viewer now toggles the value of that parameter to the other value. This provides many useful behaviors.** The sample report demonstrates toggling between viewing Profit or Revenue by simply clicking on the {?Metric} parameter. Another parameter allows toggling between expanding all top nodes or manually selecting which nodes to expand.

- **If a user clicks on a single-value date parameter (placed on the report layout), DataLink Viewer now presents an enhanced date entry/selection dialog**.

- **If a user clicks on a single-value parameter with less than 2 default values, where custom values are allowed, DataLink Viewer now presents a simple InputBox dialog, making it easier and faster for the user to change the parameter value**.

- **If a user clicks on a single-value parameter with more than 2 default values, where custom values are not allowed, DataLink Viewer now presents a multi-row dialog allowing the user to quickly select and change the selected value for that parameter.** Note: the user can simply double-click one of the listed values.

- **DataLink Viewer now allows you to click on a formula in order to change the value of a related parameter**. The formula name must start with "**DLV_Param_Click:"** followed by the parameter name you wish to change when a user clicks on that formula. For example, **"DLV_Param_Click:Customers"**. The formula itself may display any values you wish. For example, you may concatenate static text with the parameter value, or you may wish to return "No Value" if the parameter value is null, and a nicely formatted value if the parameter is not null.

- If a parameter name starts with “Password” the Selective Parameter refresh dialog now protects the value for that parameter by showing it as “*******”.

- Double-clicking a value in the special single-value dynamic parameter dialog (rendered based on a report acting as a parameter dialog) now immediately sets that value (no need to also click the OK button).

- Linked dynamic parameter reports that were saved with data during design time are now automatically refreshed within DataLink Viewer.

- Changed the user manual to pdf format with live links from the Table of Contents.

- ♦ **For report the use native connection to MS Access or Excel files, you can now let the user select the location of the database file on their machine.** This is useful when the database file location has changed or when a consultant develops reports for many different clients. For detail, see "CHANGING FOLDER LOCATION FOR ACCESS/EXCEL FILES."

- ♦ DataLink Viewer XI now uses a newer version (Service Pack 6) of the Crystal XI R2 runtime components. Users of prior XI versions can't use the online update option and must go through a Remove & Install process in order to update their version. Note: this update brings back the missing Cancel button in parameter dialogs.

- ♦ Added proper handling for a new "Database connector error" that has been introduced by newer runtime components.

- ♦ Fixed a problem related to sharing of linked dynamic parameter values with earlier regular parameter values in the main report.

- ♦ Parameter values passed via command line arguments can now be "shared" with linked dynamic parameters.

- ♦ Added a command line argument to allow scheduling and automation of rpt to rpz file conversions. For detail, see the section on "MAKE_RPZ Command Line Argument"

- ♦ Parameter dialogs in DataLink Viewer XI now **automatically position the cursor inside the first parameter**. This allows the user to start entering a parameter value without first clicking or tabbing into the dialog.

- ♦ **A new ViewMode command line argument allows you to remove the toolbar and/or the status bar from View Only report Previews** triggered from a command line with –v argument. This is **particularly useful in combination with Auto_Refresh because it allows you to maximize the space available for viewing the report**.

Version 5.9001.0: Released May 08, 2009

- ♦ **If a user clicks on a parameter (placed on the report layout), DataLink Viewer now prompts the user just for that parameter. That makes Crystal reports more interactive and more intuitive for the user.**

- ♦ **Added options to log report-use information to any database via an ODBC. This provides a mechanism for not only monitoring report performance and use, but also for addressing data privacy and protection requirements**. Recorded information includes Report Path & Name, Windows User ID, Machine ID, Start/End Time, Parameters, SQL, Connection Properties, Printing & Exporting Indicators. For detail, see "Monitoring DataLink Viewer Use" and "Record Processing to an ODBC Database."

- ♦ Added sample report demonstrating interactive features of DataLink Viewer:
  - **Click to change a single parameter**
  - **Click to change the performance measure** (Profit, Revenue, or Discount)
  - **In-place Drill-Down**
  - **Click to Change Grouping**
  - **Click to Change CrossTab Rows/Columns**
  - **Click to change report display** (Detail, Cross-Tab, or Both)
  - **Selective Parameter Refresh**

  You may wish to use this report to study and implement these techniques in your own reports.

- ♦ DataLink Viewer XI now uses a **newer version (Service Pack 5) of the Crystal XI R2 runtime components**. Users of prior XI versions can't use the online update option and must go through a Remove & Install process in order to update their version.

- ♦ DataLink Viewer 8.5 now uses **newer versions of the Crystal 8.5 runtime components**. Users of prior 8.5 versions can't use the online update option and must go through a Remove & Install process in order to update their version.

- ♦ Fixed an issue with the right-click menu option of Preview Report (problem was limited to DataLink Viewer 9).

- ♦ Fixed an issue with remembering and sharing parameter values for same-name parameters in a chain of dynamic as well as static parameters.

- ♦ Fixed an issue with printing drill-down views in Crystal XI.

- ♦ Users can now select and convert multiple rpt files into rpz files in a single step.

- ♦ By including the text "_Linked" in the name of subreport stored procedure parameter names, you can now avoid seeing such linked parameters in the report refresh dialog.

Of course, if the parameter is not linked, you should not include "_Linked" in the name of the parameter because you want to include such unlinked parameters in the selective parameter refresh dialog. For more detail, see "Selective Parameter Refresh."

- Folder Selection for FoxPro DBF Files now also influences the data source for dynamic parameters. This should be of interest to users who take advantage of the functionality described in "**Selecting Folder Location for FoxPro DBF Files (MasterBuilder)**."

- Added code to handle cases where a user clicks on the button to 'browse for reports to open' and the last used rpt folder is not accessible to that user.

- Added functionality to capture the **machine name**, **registered company**, and **hard drive serial number** where the report is running. See "Loading Machine Information into Parameters."

Version 5.8001.0: Released 7/25/2008

♦ Improved how "view only" report windows pop on the screen. There is no longer a step where a tab is briefly visible in the background.

♦ **By naming parameters in a certain way, you can now ask DataLink Viewer to automatically load their values from ini file entries**. One possible use may be for reports where some parameters change only rarely. More importantly, this **allows report developers who sell their .rpz files in vertical markets to protect their reports from unauthorized use.** It also allows **each client to customize the reports with text elements, conditional formatting, or record selection criteria without changing the report design**. For more detail, see the sections on "Load ini Values into Parameters" and "Securing Reports against Unauthorized Use."

♦ Users can now suppress the progress popup dialogs that briefly show when printing or exporting is in progress. To turn the suppress option on, set the following entry under the [Options] section of DataLink_Viewer.ini to True:

```
Suppress_Export_and_Print_Progress_Popup=TRUE
```

♦ Fixed a problem with opening .rpz files placed in a write-protected folder.

♦ Removed unnecessary page edge display from the special DataLink Viewer dynamic parameter dialogs.

♦ Added a user manual section with detailed instructions on "Cycling through Several Auto-Refreshed Reports."

♦ Maximized auto-refresh report windows now become the top window each time they get auto-refreshed. This is useful when cycling through several maximized auto-refresh reports.

♦ Fixed a minor issue with the special in-place drill-down functionality.

♦ Released a report demonstrating how users can click to
- change report grouping at multiple levels,
- control what report sections are visible, and
- drill-down within the main report view,
all without hitting the database again. The report (for DLV 9 or XI) is available from:
www.milletsoftware.com/Download/DLV_Demonstration_11.rpt

- Update.exe, the component providing online update functionality, is no longer marked as an essential component. This means that Symantec AntiVirus may (falsely) quarantine that component without causing the headaches of automatic reinstalls. In order to retain the ability to do online updates to newer versions, you can open Symantec Antivirus, expand the Configure node, select the File System Auto-Protect node, click on the Exclusions button, click the File/Folders button, and select update.exe (in the DLV folder) to be excluded from the virus protection.

- When setting up DataLink Viewer to act as a launch pad for view only windows (see the user manual section on “Remove the Preview Tab from the Initial Window”), the primary window now maintains a generic title of “DataLink Viewer”. This facilitates navigation among multiple minimized windows.

- When launching reports to a view-only window by double-clicking the grid or via the right-click menu option, the Last_Used column for the report row in the grid now gets updated.

Version 5.7001.0: Released 2/26/2008

- DataLink Viewer XI now uses a **newer version of the Crystal XI R2 runtime components**. This fixes an issue with numbers showing up as scientific notations in charts. Users of prior XI versions can't use the online update option and must go through a Remove & Install process in order to update their version.

- Added an option to the special DataLink Viewer Integrated Authentication functionality **allowing an administrator to securely share integrated authentication information with any other user who successfully logged in to the same machine**. This avoids the need to assign a different user id & password to each user and for each user to go through the initial "Remember Me" login step. For detail, see the user manual section on "Shared Machine Authentication."

- Fixed an issue related to closing a report in a View Only mode.

- Fixed a rare issue causing unnecessary logins.

- Fixed an issue with the Display_DSN_List_in_ODBC_Login option (in the ini file).

- Fixed an issue causing a change in ODBC DSN to be ignored for a report that is already connected to SQL Server.

- Fixed an issue causing the user manual and some .rpt files to be handled as essential installation files. If you now elect to delete these files, they will not be recreated automatically from the install (.msi) file the next time you start DataLink Viewer.

- DataLink Viewer XI now shows the close drill-down button only when a drill-down view is open.

- Users can now delete all report rows from the grid without causing the ReportList.txt file to be deleted (& recreated from the install file when the applications restarts).

- The **Reload button now triggers a Preview of the reloaded report**.

- **Improved handling of cases where File Locations specified in the DataLink_Viewer.ini are pointing at disconnected/inaccessible mapped drives**. In the past, such cases could cause DataLink Viewer to hang in memory. Now, the user gets a detailed message before DataLink Viewer closes.

Version 5.6001.0: Released 9/23/2007

- **Users can now select which parameters they wish to refresh**. The parameter refresh choices are stored for each report and can be easily reused or changed at a later session. Note: independent (unlinked) subreport parameters participate in this process and are listed as [subreport name] -> Parameter Name. For detail, see "Selective Parameter Refresh."

- **Fixed an issue with .rpt-to-.rpz conversion on Vista machines**.
  <u>Important Note:</u> older .rpz files can be opened by this new version. However, if you package new .rpz files, users would need to install the new version of DataLink Viewer in order to run them. To ease the transition, you may wish to retain one PC with the old DLV version and package the .rpz files on that machine.

- DataLink Viewer XI now uses the Crystal XI **R2** (Release 2) runtime components. This provides a couple of nice export features:
  - the **pdf** export dialog provides an option to "**Create bookmarks from group tree**"
  - the **Excel (Data Only)** export dialog provides an option of "**Show group outlines**," which allows you to expand and collapse group detail in excel.
  Users of prior XI versions must go through a Remove (using old msi file) & Install.

- Fixed an issue causing the ODBC DSN selection option in the login dialog to not appear in some cases.

- Fixed a login issue when previewing a report with saved data and then refreshing it.

- Fixed an issue causing reports that use two different data sources: one with a password and the other one a DAO connection to MS Access without a password, to trigger a login dialog for the Access data source.

- Updated user manual section on "Launch a Report While Viewing Another Report".

- Added code and a message to handle cases where the ReportList.txt file location option is pointing at an invalid directory.

- The report listed in the first row (Select Report tab) can now be selected and launched by clicking the Preview tab after a single click.

- Added a DataLink_Viewer.ini option to remove the Set To Null checkbox for date or datetime parameters when using Stored Procedures as a data source. To turn this option ON, you need to set the following entry in **DataLink_Viewer.ini** to TRUE:
  [Options]
  Disable_Stored_Procedure_Null_Checkbox_for_Date_Parameters=TRUE
  Note: this option **applies only to the CR 9 version of DataLink Viewer** since the other versions don't need it.

- ♦ In cases where you have both a “master” DataLink_Viewer.ini (in the application folder) as well as a “user” DataLink_Viewer.ini file (in a different folder), **you can now specify which ini file, “master” or “user” should control certain options**. For detail, see “Enforcing Settings in a Master DataLink_Viewer.ini File.”

- ♦ Added an option to the special DataLink Viewer Integrated Authentication functionality allowing all Windows authenticated users to share an encrypted global password (the database user id is automatically set to the currently logged in Windows user id, but a secret global password is shared across all users). This avoids the need to assign a different password to each user and for each user to go through the initial “Remember Me” login step. Instead, **an administrator can activate Integrated Authentication (and periodically change the global password) for all users in the organization in a single step**. Contact MilletSoftware and, if your use scenario seems to fit this feature, you will receive instructions.

- ♦ The special DLV_User_ID parameters now refreshes when a report with saved data is refreshed by another user.

- ♦ Implemented a more complete blocking of exports when the Disable_Export_Button option in the DataLink_Viewer.ini file is set to true. In such a case, the right-click menu from the report list no longer shows an Export menu item. Also, attempts to export from the right-click menu in file explorer are blocked with a message indicating that exporting has been disabled by the administrator.

- ♦ Added an **option for hiding the Preview tab** in the initial DLV window -- **turning the main DLV interface to a simpler launch pad for multiple report preview** -- and updated the user manual to provide a review of various ways to launch reports into new view only windows For detail, see “Launching Reports in a New Window.”

- ♦ Faster setup processes when loading and refreshing reports.

Version 5.5001.0: Released 1/12/2007

- Added an option to **convert double-clicks from regular drill-down action to In-Place Drill-Down action**. For detail, see: "Converting Section Double-Click to In-Place Drill-Down."

- **Added compatibility with document management programs such as Documentum**. Users of such programs can now select a Crystal Report document and have it open within DataLink Viewer.

- **Added code to avoid Outlook security warning messages when exporting to MAPI (email)**. To enable this functionality, set the following DataLink_Viewer.ini option: Control_Outlook_Security_On_Export=True

- Added a message box to block attempts and clarify that .rp**z** files can't be exported to MAPI in a single step. The user is prompted to first export to disk, and then manually email the resulting file. Note: regular .rp**t** files don't have that restriction. This limitation (due to a bug in the Crystal runtime components) applies only to the 9 & XI versions of DataLink Viewer (not to the 8.5 version).

- Fixed an issue in the 8.5 and XI versions that caused "view only" previews to not remember the window size and position.

- Fixed an issue with launching one report from another (double-clicking a section with DLV_RUN or DLV_RUN_HERE formulas). The problem was that in some scenarios an unexpected drill-down action was automatically executed on the launched report.

- Fixed printing issues in the Crystal XI version of DataLink Viewer.

- Enhanced handling of invalid or renamed .rpz files.

- **Added options to remove the Print, Export, Select Expert, and Search Expert buttons in the viewer**. This can be done via either ini file entries or command line arguments. This allows, for example, an application that launches a report in DataLink Viewer to control which users are allowed to print or export the report. For more detail, see the "Disabling Print, Export, Select Expert, and Search Expert Buttons" section.

- **Added a Connect_To_SQLOLEDB command line argument for selecting a different SQL Server when using OLE DB as a connection method**. For detail, see "Selecting an Alternative SQL Server – OLE DB Data Source."

Version 5.4001.0: Released 7/21/2006

- Fixed an issue related to clicking on On-Demand subreport.

- Added an option (See Options Dialog, Parameters Tab) to avoid the confirmation dialog (OK, Cancel, Auto-Refresh) in cases where the user refreshes a report that doesn't have parameters or doesn't require a parameter re-prompting (Options set to reuse parameter values on Refresh). By turning this option on, users are no longer bothered with the confirmation dialog and the report gets refreshed immediately upon clicking the Refresh toolbar button.

- After refreshing a report, DataLink Viewer now keeps the same page number. If the user was on the last page of the report, DataLink Viewer displays the last page of the refreshed report.

- **Added DataLink_Viewer.ini options for restricting user access to only the reports listed in the grid** (the user can't browse for other report files by clicking the File Open button), disabling the User Manual button, and disabling the Version Information button:

  ```
  [Options]
  Disable_Browse_Dialog=True
  Disable_User_Manual=True
  Disable_Version_Info=True
  ```

  Note: when disabled (set to **True**), when a user clicks on the button, a message explains the restriction and disables the button.

- Added DataLink_Viewer.ini option for forcing ODBC login dialogs to always show the list of alternative ODBC DSNs. The default, is to show only a button that, when clicked, expands the dialog and shows the list of ODBC DSNs.

  ```
  [Options]
  Display_DSN_List_in_ODBC_Login=True
  ```

- Added the ability to embed user input in command line calls from one report to another (DLV_Run). One use scenario is invoking label printing for a given product but allowing the user to control the number of copies to print. For detail, see “Embed Input from The User in the Command Line Call for Another Report.”

- Fixed an issue with directing subreports to a different ODBC DSN when main report doesn't require login (e.g., SQL Server Integrated Authentication).

- The installation of DataLink Viewer now adds a registry key indicating the path to the executable.
  The key is HKEY_LOCAL_MACHINE\SOFTWARE\Classes\<DLV EXE Name> and the property name is "path". This key is removed upon an uninstall.
  This is useful for other applications that invoke DataLink Viewer via command lines.

Version 5.3001.0: Released 5/24/2006

- When clicking the Refresh button, the dialog allows you to auto-refresh the report every N seconds. The auto-refresh process stops when the user clicks the refresh button again or when a new report is selected. You can also trigger auto-refresh by using an "Auto_Refresh" command line argument. For detail, see "Auto-Refresh Reports Every N Seconds".

- Added an option (see Options dialog) to control what happens when a user clicks the Refresh button and the report has parameters: **RePrompt** for new parameter values, **ReUse** the current parameter values, or **Ask** the user (the default behavior).

- **Added In-Place Drill-Down functionality**. This allows users to expand and collapse details within a single report tab. For detail, see "In-Place Drill-Down".

- Added the ability to react to clicks on one formula by setting the value of another formula. Among other things, this allows users to re-sort reports by clicking on column headers. For detail, see “Click to Change Parameter Value”, “Click to Set Formula Value”, and “Click Column Headers to Resort the Report.”

- Various improvements to printing via a command line: 1) added a **Print_Copies** argument to allow scheduling or triggering of report printouts with a specified number of copies, 2) added a "**Printer:Dialog**" argument for requesting printing with a printer selection dialog, 3) specifying "**Printer:Default**" now works even when the report was saved in Crystal with a "No Printer" option, 4) fixed a duplicate printing problem in the Crystal 8.5 version of DataLink Viewer.

- Fixed a problem causing an error message when printing reports to a specified printer when the report uses a custom paper size.

- Fixed an exporting to Excel 97 (Data Only) problem when using .rpz files in the Crystal 9 version of DataLink Viewer.

- The **Strip_Table_Qualifiers option now applies even for cases with table aliases**.

- A forced login for main reports no longer forces login for dynamic parameter reports.

- Fixed sharing of parameter values across linked dynamic parameter reports.

- Added an "ODBC_DSN_From_To" command line argument.
  This is useful for cases where a report uses multiple ODBC DSNs and you need to override only one of them. .

- When selecting a new ODBC DSN in the login dialog for a report, that **DSN choice carries over to any dynamic parameter reports** used to provide parameter values for the report (if they use the same original ODBC DSN).

- A new ini file option allows you to restrict the choice of alternative ODBC DSNs (in the login window) to a specified group or groups of ODBC DSNs. For detail, see "Restricting the List of ODBC DSN Choices".

- Added an option to avoid the printer setup dialog in DataLink Viewer 8.5 and 9.
  If you set the following DataLink_Viewer.ini entry to False:
  Prompt_for_Printer_Setup=False
  when you click on the Print button, the Printer Setup dialog is skipped and you go directly to the dialog controlling the number of copies. This is a good approach when you know the user will always use the default printer.
  Note: in DataLink Viewer XI those 2 dialogs are combined into a single dialog.

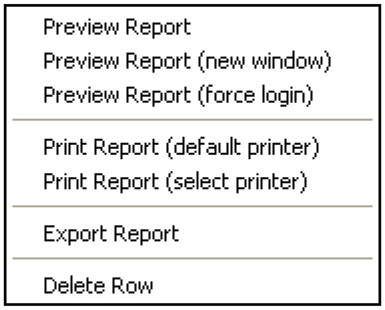
- Windows **File Explorer right-click menu options now include DataLink Viewer Preview, Print, and Export actions for .rpt/.rpz files**. For detail, see "Launch Reports from File Explorer".

Version 5.1001.0: Released 11/11/2005

- A new command line argument ("**Printer**:Default" or "**Printer**:Printer_Name") allows you to print a report without previewing it. This allows you to **trigger or schedule printing of Crystal reports** using other applications or the free Windows Task Scheduler.

- A new command line argument ("**Export:Dialog**") allows you to start the export process for a given report. While this doesn't allows scheduling of report exports (for that you would need a tool such as my Visual CUT software), it allows other applications to trigger the process to the point where the user is presented with the export dialog (without the need to select or preview a report. For detail, see **Error! eference source not found.** on page **Error! Bookmark not defined.**.

- The pop-up menu shown when right-clicking a report row in the report list grid now provides **options for exporting and printing without previewing the report**. This is useful in cases where the user wishes to send a long-running report to the printer and doesn't wish to wait for the report to preview before sending it to the printer.


Preview Report
Preview Report (new window)
Preview Report (force login)
Print Report (default printer)
Print Report (select printer)
Export Report
Delete Row

- Added a DataLink_Viewer.ini option to always force a login dialog. **This is useful in cases where the users should always specify an ODBC DSN or a Database name.** To turn this option on, you need to set the following entry in **DataLink_Viewer.ini** to TRUE:

```
[Options]
Always_Force_Login=TRUE
```

- Fixed an issue causing the ODBC_DSN command line argument to be ignored in some cases.

- Fixed a window size issue when opening reports using the view only option.

- To address a rare scenario under Citrix and Windows 2003, an ini file option was added that, if set to TRUE, causes an explicit assignment of the default printer when the user clicks the print button (before the printer selection dialog is displayed):

  ```
  [Options]
  Set_Report_To_Default_Printer=TRUE
  ```

- The functionality of ***Selecting Folder Location for FoxPro DBF Files*** now applies even to reports created with Crystal versions prior to Crystal 9.

- Added the ability to **control the default values of multi-value dynamic parameters via a formula within the dynamic parameter report**. This was previously possible only for single-value dynamic parameters. For detail, see Controlling the Default Value in Dynamic Parameters on page 84.

- Added the ability to **force the user to select a value in a dynamic parameter dialog, before clicking the OK button.** For detail, see Requiring a Value for a Dynamic Parameter on page 85.

Version 4.8001.0: Released 8/12/2005

- Fixed a problem with launching another Crystal report into the same Preview window (using the DLV_Run_Here formula approach).

- Improved the centering of dynamic parameter dialogs within the main screen.

- Added a DataLink_Viewer.ini option to **set the default value of String parameters to an empty string (""). This applies only for Discrete, Single-Value, String parameters that were assigned no default values.** To turn this option ON, you need to set the following entry in **DataLink_Viewer.ini** to TRUE:

  ```
  [Options]
  Set_String_Parameters_Default_To_Empty_String=TRUE
  ```

  If your report has more than one parameter, by setting this option to TRUE, **the user no longer has to select (click on) each string parameter to ensure its value is not left as Null.**
  Note: this option **applies only to the CR 9 version of DataLink Viewer** since the CR 8.5 & CR 11 runtime component versions behaves that way by default.

- Added a command line argument option for specifying that a parameter value is Null as well as a new section in the user manual about setting parameter values via command line arguments.

- You can now **control the default selected for single-value dynamic parameters via a formula within the dynamic parameter report**. For example, you can have the last customer order in the dynamic pick list be the default value. For detail, see Controlling the Default Value in Dynamic Parameters on page 84.

- Reports now remember their Group Tree display status.

- **Right-clicking the report list provides a new menu option for previewing the report in a "view-only" new window.** This is useful when you wish to have multiple reports open at the same time. To support cases where users wish to have this as the default launch mode, **a new option under the *Launch* tab of the Options dialog allows you to indicate that double-clicking a report row should launch the report into a "view-only" window**.

- **Reports that are displayed in a "view-only" window remember their window size, position, zoom, and group-tree status**. These settings are reapplied when the same report is opened again in a view-only window.

- Clicking Cancel from parameter dialogs when launching a report to a view-only window now closes the window immediately.
- For ODBC data sources, users can now directly enter a database name into the login dialog if they wish to override the database specified in the ODBC data source or in

the report itself.

- A new command line argument (Oracle_Server) allows external control of which Oracle server is used with the report you are launching.

Version 4.7003.0: Released 4/27/2005

- Fixed a connection mode issue when dynamic folder selection is used for Visual FoxPro free (.dbf) tables.

- When using a Native connection to Oracle, **you can now edit the Server name in the login dialog and run the report against a different server**.
  Thanks to Mark Gregory (Toronto Star) for sponsoring this development.
  Note: This functionality is not available in the DataLink Viewer version that uses Crystal 8.5 runtime components.

- In previous versions under certain situations (e.g., Oracle connections), the Preview with Forced Login option didn't allow changes of database source once the report was connected to its original data source. Users had to close and restart DataLink Viewer in order switch the same report to a different data source. This limitation has now been removed.

- DataLink Viewer now **supports selection of a new ODBC data source even when the database name is different from that used in the original ODBC DSN and even when both databases reside under the same server** (e.g., different databases under a single SQL Server). To ensure this works even when the table names in your report are fully qualified (database.owner.table), set the following line under the [Options] section in DataLink_Viewer.ini to True:
  ```
  Strip_Table_Qualifiers=True
  ```

- Fixed an error when a user cancels out of a parameter dialog.

- Fixed a problem in changing ODBC DSN sources when running the Crystal 9 runtime components version with reports from earlier versions of Crystal.

- Parameter values (even if they are static parameters) are now automatically reused if the same parameter name is used more than once in a cascading parameter situation. For detail, see Sharing Values Across Cascading Parameters on page 83.

- The installer(msi) file now **defaults to a per-machine installation, if installed by an Administrator**. This facilitates the deployment of DataLink Viewer when an Administrator installs the software for use by other users. In the past, such cases would require installation from a command line with an ALLUSERS=1 argument.

- DataLink Viewer 9 is now using a new set of Crystal 9 runtime components that, among other minor fixes, can provide **much faster execution of Crystal 8.5 reports** in certain cases (one user reported a decrease from 3-5 minutes to 10 seconds).

Version 4.5003.0: Released 1/28/2005

- Reports that were designed to not have a Group Tree no longer cause the Group Tree button in the viewer toolbar to disappear. Thanks to Peter Frank (Canadian Museum of Nature) for identifying the issue.

- Fixed a printer selection problem (introduced by code changes in the 4.5001 release).

- When selecting rpt files from the file system, DataLink Viewer now remembers the last used folder (by storing it in the ini file).

Version 4.5001.0: Released 1/5/2005

- When running reports that use ODBC data sources, you can **select which ODBC data source should be used**. This is explained in detail in the section: Select Alternative ODBC Data Sources for the Same Report on page 40.

  Thanks to Dan Macdonald (Royal Canadian Mounted Police) for sponsoring this development.

- **Right-clicking the report list provides a new menu option for forcing a login window before previewing the report.** This is useful when you wish to run the same report against another ODBC data source, when you wish to login to the same data source under a different user id (without restarting DataLink Viewer), and when you wish to temporarily avoid the default login information provided by Integrated Authentication.
  This is explained in more detail in the section: Forced Login on page 45.

  Note: if you use Oracle ODBC connections, there are some scenarios where the only way to avoid connecting to a previously opened connection is to close and reopen DataLink Viewer.

- **A new command line argument (ODBC_DSN) allows external control of which ODBC data source is used** with the report you are launching from a command line. See page 22 for more detail.

- Fixed a failure to save some text changes in the report list.

- Improved printer selection functionality.

- Integrated Authentication is now stored in the ini file with a key for not only the user but also the PC. This allows using a centralized DataLink Viewer.ini file where the integrated authentication information from many users and many PCs is stored only once.

- **.rpt & .rpz file extensions are now automatically associated with DataLink Viewer** (if they are not already associated with another application). When starting DataLink Viewer, if the .rpt/.rpz file extensions are not already assigned to another program, these extensions are assigned to the DataLink Viewer executable (so a double-click on a .rpt/.rpz file would open the report in DataLink Viewer). This means that users no longer need to manually specify the Open With option in the Windows File Explorer.

Version 4.3004.0: Released 11/22/2004

- Added **Integrated Authentication** functionality. This allows users to automatically connect to any secure database, provided they first: a) turn on the Integrated Authentication option, and b) log in once to the secure database using a new "Remember Me" option. The database login information is stored in DataLink Viewer.ini in a **strongly encrypted format**. DataLink Viewer uses this information only after validating that it belongs to the **current PC and current Windows user**.

  For full detail, see **Error! Reference source not found.** on page 40 of the user manual.

  Thanks to Peter Peterson and Gary Anderson (Boulder County) for sponsoring the development of this functionality.

- Fixed an issue related to storing the rpt files in a folder with no delete permissions for the users.

- Fixed an error message when closing DataLink Viewer after deleting all reports from the report list.

- A check for ODBC DSN is now not case sensitive.

- Double-clicking on sections with a string formula for launching another executable (using "**EXE_Run:executable:arguments**") now works even if the specified executable contains a "**:**" due to a specified path, such as c**:**\…

- When opening *.RPZ files (encrypted reports), DLV now populates report title and subject information (if specified during report design) into the grid.

- You can now **protect ReportList.txt** (the content of the report list in the 1st tab grid) and/or **ReportList.grd** (the style and layout of the grid) from changes by setting the attributes of these files (or their folders) to Read Only. This is useful in cases where an administrator wishes to maintain the report list content for all users (not allowing individual users to add or delete reports). In past versions, such situations would cause an error message when closing DataLink Viewer.

Version 4.1001.0: Released 8/28/2004

- Fixed issues related to launching reports using command line arguments and logging use statistics.

Version 4.0000.0: Released 8/17/2004

- **New Grid** for Selecting Reports with enhancements such as:
  - **more columns** (path, report, title/description, **subject**, **type**, date/time **last used**)
  - **sort on any column by clicking the column header**
  - **group on any column by dragging the column header to a Grouping area**
  - **select larger or smaller font size** (right-click column headers…)
  - **hide/show columns** (right-click column headers…)
  - Select hundreds of reports and load them into the grid in one step

  Thanks to Dave Leland (Johnson Corporation) and Paul Callaghan (Larsen Hinge) for sponsoring this development.

- Fixed an issue with the extra path option for dynamic parameter reports.

- The viewer now **shows an animation while the report is loading**.

- The **main window remembers its size from session to session**.

- The **login window now shows the Server name and Database name.**
  Thanks to Doug Pichler (National Business Institute) for sponsoring this development.

- **Added a message box alerting the user to cases where the ODBC DSN (Data Source Name) expected by the report is not set up on the PC**.

- Added a DataLink_Viewer.ini option to **set the default value of Date/DateTime parameters to TODAY's date (only for Discrete, Single-Value parameters that were assigned no default values).** To turn this option ON, you need to set the following entry in **DataLink_Viewer.ini** to TRUE:

  ```
  [Options]
  Set_Date_Parameters_Default_To_Today=TRUE
  ```

  If your report has more than one parameter, by setting this option to TRUE, **the user no longer has to select each date parameter to ensure its value is not left as Null.**
  Note: this option **applies only to the CR 9 version of DataLink Viewer** since the CR 8.5 runtime component version behaves that way by default.

- Added DataLink_Viewer.ini options to **set 3 lines in the About dialog to text of your choice.** This is useful if you sell Crystal reports bundled with DataLink Viewer and you wish to specify your contact information for technical support. For detail, see **"Add your Company Info to the About Dialog."**

- Added an option for **logging use statistics**. If you specify (in the Options dialog) a folder location for a **DLV_Use_Log.txt** file, use statistics (User_ID, PC Name, Report, Date & Time Started the report loading, Date & Time Report Finished Loading) are appended to that file. This is particularly useful for Citrix/TS installations where you can use that file (for example, using Crystal with a Text ODBC driver) to analyze which reports are popular, who runs which reports, and which reports take too long to run.
  Thanks to Dave Leland (Johnson Corporation) for sponsoring this development.

Version 3.3971.0: Released 6/22/2004

- Added **DataLink_Viewer.ini** options for **restricting user access to certain dialogs**:

  ```
  [Options]
  Disable_RPZ_Creation=False
  Disable_Check_for_Updates=False
  Disable_Options_Dialog=False
  ```

  Note: when disabled (set to **True**), the user interface doesn't reflect that fact until the user actually attempts to access the dialog. Thanks to David Leland (Johnson Corporation) for suggesting this functionality.

- The dialog for selecting rpt/rpz files from the file system now **allows you to select multiple reports and load all of them into the DataLink Viewer grid**.
  Thanks to Jon Newsom for suggesting this enhancement.

Version 3.3951.0: Released 6/10/2004

- Removed the control box **[x]** at the top right corner of Dynamic Parameter dialogs. This ensures users use the Cancel button if they wish to Cancel/Close the dialog.

Version 3.3911.0: Released 6/01/2004

- Fixed a previewing failure problem on some PCs.

Version 3.3901.0: Released 5/25/2004

- Fixed an issue with dynamic parameters behaving as static parameters if the user clicked Cancel out of a dynamic parameter dialog and then refreshed the same report.

Version 3.3801.0: Released 5/12/2004

- Fixed an error message triggered when a command line invocation of DataLink Viewer had trailing spaces.

Version 3.3701.0: Released 4/23/2004

- ♦ Fixed an error message triggered when a user clicked Cancel in a standard (non-dynamic) Crystal parameter dialog if the report was running against a secure database. Note: this problem occurred only in the Crystal 9 runtime components version of DataLink Viewer.

Version 3.3601.0: Released 4/19/2004

- ♦ Fixed an error message ("A form can't be moved or sized while minimized or maximized") caused by starting DataLink Viewer on a small or low-resolution screen. Note: applies only to the Crystal 9 runtime components version of DataLink Viewer.

- ♦ Added a new **DLV_Run_Here** option for launching a second report by double-clicking a section in a first report and opening the second report within the current preview tab. The previous option (DLV_RUN) still launches the second report in a new instance of DataLink Viewer. These options are described in the ***Launch a Report while Viewing another Report*** section of this user manual.

Version 3.3401.0: Released 3/27/2004

- ♦ The component dealing with online updates for the software has been enhanced and a minor problem with online updates under NT4 PCs has been fixed.

- ♦ If a user selects a new folder location for the File DSN data source, dynamic parameter reports automatically use that same new data source location (if the dynamic parameter reports use the same File DSN).

Version 3.3301.0: Released 3/21/2004

- ♦ **Removed the "Preview" tab at the top of the main report viewing area to avoid wasting screen space.** If a drilldown view is activated by a user, both the "Preview" tab as well as the drilldown tab become visible to allow user navigation between these views. The same functionality also applies to dynamic parameter dialogs.

Version 3.3101.0: Released 3/08/2004

- Fixed a refresh (same parameter values) issue for encrypted (rpz) reports.

- The CR 9 runtime components version of DataLink Viewer now support dynamic selection of data folders for reports using the Visual FoxPro ODBC driver for a File DSN (using “Free Tables”, which are dbf files under a given folder).

  The development of this functionality was suggested and sponsored by Jon Newsom (Newsom & Associates) in order to support ***Intuit Master Builder*** company reports where each company’s data is located in a different folder on the same PC).

  For full detail, see Selecting Folder Location for FoxPro DBF Files (MasterBuilder) on page 46 of the user manual.

Version 3.2001.0: Released 3/02/2004

- Fixed an issue causing refreshes of reports with no parameters to “hang” under special circumstances. Thanks to David Leland (Johnson Corporation) for identifying the issue.

Version 3.1501.0: Released 2/27/2004

- **You can now specify User ID and Password as Command Line arguments when simply starting DataLink Viewer** (without specifying a report to view). This is useful in cases where you wish to avoid prompting the users for login information when they select reports to view.

  A typical scenario would be to create a desktop shortcut and specify, as the Target property of the shortcut, a command line such as:
  "C:\Program Files\DataLink Viewer\DataLink_Viewer.exe" "User_ID:dba" "Password:sql"

  Note: exercise care when using this feature because, in the shortcut case, the password is visible to anyone who can gain access to the shortcut properties.

  Thanks to David Leland (Johnson Corporation) for suggesting, sponsoring, and testing this new functionality.

Version 3.1001.0: Released 2/18/2004

- Users can now leave multi-value dynamic parameters empty (no rows selected) and DataLink Viewer would set the parameter to “” (for String parameters) or Null (for all other data types) without triggering the default Crystal parameter dialog.

  Thanks to John Daniell (Positive Business Solutions) for identifying the issue and testing the fix.

Version 3.0.00: Released 2/11/2004

- **Added a button in the version dialog allowing users to check for software updates on my web site and apply those patches quickly and efficiently**.
  The patch sizes are typically very small since this automated process applies only net changes to files.

- By default, dynamic parameter dialogs for single-value parameters allow the user to enter a value directly instead of clicking on the displayed list. **You can now force the user to select from the list displayed by the dynamic parameter report** under two scenarios:
  1. You include the text '**NoEdit**' in the parameter name.
  2. The user double-clicked an item in the dynamic parameter report and that item has an associated data value (a formula named Parameter_Data). Note: if the option to remember dynamic parameter values is left on, from that point on that parameter will force the user to select from the list.

  Thanks to Abbas Rostami (Cubic Corporation) for suggesting these changes.

Version 2.9.60: Released 1/26/2004

- **Added a button to the 1st tab for launching an Options dialog (instead of manually editing DataLink Viewer.ini).**

- **Dynamic parameter dialogs (for each dynamic parameter report) now remember how the user last modified** them in terms of **Size** (width & height), **Group Tree** (on or off), and **Zoom** level.

- Fixed an issue with sharing (reusing) dynamic parameter values among multi-value parameters when such parameters are used repeatedly for the same main report or its dynamic parameter reports.

- ***DataLink Viewer* now remembers the last User_ID used to login to the database**.
  This is helpful in typical cases where the developer's User_ID (saved within the report during report creation) is not the User_ID for the person running the report in DataLink Viewer.
  Note: to disable this option, edit DataLink_Viewer.ini and set the **Last_User_ID** entry under the **[Options]** section to "DLV_Disabled"
  [Options]
  Last_User_ID =DLV_Disabled

Version 2.9.30: Released 1/08/2004

- **Fixed a Unicode-related problem in opening an RPZ file in DataLink Viewer when running on a PC using a non-USA character set.**
  Thanks to Michael Taupp (T-Online) for identifying the problem and testing the fix.

Version 2.9.20: Released 12/07/2003

- **Fixed a bug causing refresh of a previewed report to not show new data in the database if the user didn't specify new parameter values.**
  Thanks to Abbas Rostami (Cubic Corporation) for identifying the problem and testing the fix.

- Added an ini file **option for specifying a path to a shared location of dynamic parameter reports**. This can be useful in cases where you deploy reports to different folders based on user permissions but wish to place dynamic parameter reports in a shared location.

  The search order for dynamic parameter reports is now:
  1. Folder where the main report resides
  2. The DataLink Viewer application folder
  3. The folder specified in a new ini file option. For example:
  ```
  [Options]
  Dynamic_Parameter_Reports_Path=V:\Crystal\Shared_Reports\
  ```

  Thanks to Abbas Rostami (Cubic Corporation) for suggesting this enhancement.

- **Added sample reports designed to work with the Crystal 9 Extreme Sample Database** (ODBC data source named: "***Xtreme Sample Database 9***"). The previous sample reports are still included and they work with the "***Xtreme Sample Database***" that gets installed with prior versions of Crystal.

Version 2.9.00: Released 12/01/2003

- Ensured the new button for launching the user manual is kept invisible when launching a report from a command line with a View Only (-v) option.

- When the user clicks the Refresh button **DataLink Viewer now lets the user indicate if the parameter values should be reused or re-prompted**.
  If the user indicates the data should be refreshed with the same parameter values, the viewer maintains the same page position. This is standard Crystal behavior.

- **You can now specify what is the default action when DataLink Viewer opens a report with saved data.**
  The default action remains "**Prompt**" (asking the user if the data should be refreshed). You can now avoid prompting the user by specifying "**Display**" (show the report with its saved data without asking the user) or "**Refresh**" (refresh the report data without asking the user).
  You specify these options by entering ***Refresh***, ***Display***, or ***Prompt*** as the values for the ***Saved_Data_Action*** entry under the **[Options]** section of **DataLink_Viewer.ini**
  For example:

  ```
  [Options]
  Saved_Data_Action=Display
  ```

- **You can now specify what is the default action when a user double-clicks a report file if the rpt or rpz file extension had been associated with the DataLink Viewer executable.**
  The default (***View and Select***) is to display both report selection as well as the preview tabs (so the user can select other reports from the 1st tab).
  Alternatively, you can **restrict the user to just view the double-clicked report** by entering ***View Only*** as the value for the ***Associated_File_Launch_Mode*** entry under the **[Options]** section of **DataLink_Viewer.ini**
  For example:

  ```
  [Options]
  Associated_File_Launch_Mode=View Only
  ```

- When launching a report by double-clicking a report section with a launch formula (see the section on ***"Launch a Report while Viewing another Report"***), **the User_ID and Password from the launching report are now automatically applied to the launched report**. This removes the need for multiple logins in such a scenario.
  You can still provide User_ID and Password arguments in the launch formula if you need to override the login information from the launching report.

Thanks to Abbas Rostami (Cubic Corporation) for suggesting and testing these changes.

Version 2.8.50: Released 10/11/2003

- Added a **button for launching the user manual** from the 1st Tab.

Version 2.8.40: Released 9/24/2003

- Fixed an issue, introduced by the prior release, causing DataLink Viewer to not display itself immediately (with an hourglass) while busy loading a report from a command line.

- Fixed an issue causing DataLink Viewer, when invoked from a Command Line, to not disappear when closed.
  Thanks to Scott Stanek (Arr-Maz Products LP) for identifying the issue and testing the fixed version.

- Improved handling of situations where users click “Cancel” from parameter dialogs.

- Fixed an issue with displaying DataLink Viewer on low-resolution displays (at or below 800 x 600). Thanks to Kevin Longley (Citronics) for detecting the problems and testing the fixed version.

Version 2.8.00: Released 8/27/2003

- Added functionality to **protect report designs** by allowing users to **Compress and Encrypt rpt files into rpz files. Users can run the resulting rpz files in DataLink Viewer, but cannot view or modify them in Crystal Reports.**

  Within DataLink Viewer, rpz files behave just like rpt files except that, in order to protect the report design, exporting rpz files to "rpt" format or to "report definition" format is blocked by DataLink Viewer. For full detail, see the new user manual section on “Protecting Report Designs with rpz Files”.

- Provide “cleaner” response in situations where a user clicks “Cancel” from Cascading Dynamic Parameter as well as from Static Parameter dialogs.

- Handle cases where a user manually edits the **ReportList.txt** file, leaving it in a format that is unacceptable for the report list (shown in the 1st tab grid). DataLink Viewer now simply fixes the file format.
  Note: manual editing of **ReportList.txt** is typically not needed now that the user can right-click the grid and use the popup menu option to delete a report row.

- Added missing dll’s to the install in order to support direct (non-ODBC) connection to DB2 databases. Note: this applies only to the CR 8.5 runtime components version of DataLink Viewer. Most users of DataLink Viewer are running the CR 9 version and the Install for that version always included those dlls.

Version 2.7.00: Released 7/25/2003

- New User Interface in the Select Report (1^{st}) Tab:
  1. A grid shows three columns: **Report Path**, **Report File Name**, and **Report Title** (from the summary information specified for the report during Design Time) or a user-specified **Description** (entered by the user by clicking in that cell).
  2. Right-Clicking a row on the grid provides a **popup menu** with '**Preview Report**' and '**Delete Row'** options.
  3. The **grid columns resize** when the viewer gets resized by the user.
  4. The **grid columns can also be resized by the user and "remember" their widths** from session to session.
  5. The data is still maintained in ReportList.txt and **the new version recognizes and migrates the data in an old ReportList.txt files automatically**.
  6. **Selecting a report to open via the "Browse for a Report to Open' button causes an immediate preview** (no need to select the Preview Tab).
  7. **Double-Clicking a report in the grid of previously opened reports causes an immediate preview** (no need to select the Preview Tab).

- Fixed a problem in printing Drill-Down views.
  Thanks to Nola Carr (Kileel Developments Ltd.) for identifying the problem and testing the solution.

- Compiled using recently updated runtime Merge Modules
  (applies only to the CR 9 runtime components version of DataLink Viewer).

- A **[File_Locations]** section in **DataLink_Viewer.ini** (located in the DataLink Viewer application folder) allows users to override locations for **DataLink_Viewer.ini** (stores processing options, previously selected values for dynamic parameters, and column widths in 1^{st} tab grid) and for **ReportList.txt** (stores 1^{st} tab grid information about previously opened reports).

  Here is an example of this section in **DataLink Viewer.ini**:

  ```
  ----------------------------------------------------------------------
  [File_Locations]
  ini_file=U:\MilletSoftware\DataLink_Viewer.ini
  ReportList=U:\MilletSoftware\ReportList.txt
  ----------------------------------------------------------------------
  ```

  This **allows a single installation of DataLink Viewer on a *Citrix* or *Terminal Server* to support individual settings for each user** by using the **DataLink Viewer.ini** and **ReportList.txt file** under the mapped user drive.
  Note: if the target folders don't exist, they get created. If the target files don't exist, they get copied from the master copies in the application folder (if they exist there).

  Thanks to Austin Henderson (FirstFleet) for suggesting and testing this enhancement.

Version 2.5.20: Released 6/7/2003

- While DataLink Viewer is busy retrieving a report, the "Select Report" tab is disabled to ensure the retrieval is completed without errors. Fixed a related error caused by minimizing and maximizing the viewer during data retrieval.

  Thanks to Curt Gravatt (Intercim) for identifying the problem and testing the solution.

- Added a **–V** ("***View Only***") processing mode option in Command Line invocation of reports. This option provides a "***View Only***" mode since ***the "Select Report" tab is not available to the user***. For example, the following syntax in a batch file, desktop shortcut, or a RUN command is using a **–V** instead of **–S** option:

  "C:\Program Files\DataLink Viewer 9**DataLink_Viewer_9.exe**" **-v** "C:\Program Files\DataLink Viewer 9**DataLink_Viewer_Year_and_Product_Prompts.rpt**" "**Parm1:1997**"

  The report is then shown with no options to select and open another report:
  Thanks to Dick Brown (Coldwell Banker) for suggesting and testing this new feature.

Version 2.5.00: Released 5/25/2003

- DataLink Viewer can now automatically limit the data displayed in a Crystal report to the records associated with the User ID logged in to the PC.
  For detail, see the section on "***Controlling Data Access according to PC Login***".

  Thanks to Dick Brown (Coldwell Banker) for suggesting and testing this new feature.

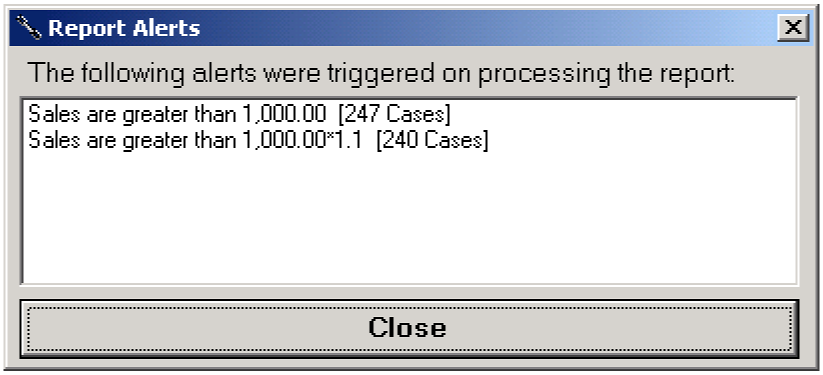
Version 2.4.11: Released 4/11/2003

- Fixed a problem with Parameters with Currency data type.

  Thanks to Helen Hiebert (Independent Consultant) for identifying the problem.

Version 2.4.1: Released 4/02/2003

- *DataLink Viewer* now displays ***Report Alerts*** as shown below:



The following alerts were triggered on processing the report:

```
Sales are greater than 1,000.00  [247 Cases]
Sales are greater than 1,000.00*1.1  [240 Cases]
```

Besides the message designed for the Report Alert within Crystal, *DataLink Viewer* adds the number of alert cases detected.

- A Close button [x] is now available to close Drill Down views.
  Thanks to Charles Martin (John H. Carter Co.) for the suggestion.

- DataLink Viewer now **also** accepts dynamic parameter names such as:
  Prompt_Products_rpt instead of **Prompt_Products.rpt**
  This allows using dynamic parameters with Stored Procedures that may not tolerate parameter names with a dot.
  Thanks to Ken Hamady (www.KenHamady.com) for the suggestion.

Version 2.3.9: Released 2/17/2003

- Upgraded runtime components
- Fixed problem related to installations (e.g., install by one user but use by another).
- **Fixed sporadic Export and Print freeze problems**.
- The sample report (DataLink_Viewer_Year_and_Product_Prompts.rpt) now demonstrates the technique of launching another report (and passing a parameter value to it) by double-clicking a report section.

Version 2.3.8: Released 1/23/2003

- Added sections to the User Manual describing how DataLink Viewer allows you to:
  1. Launch a Crystal Report from a Command Line
  2. Launch a Crystal Report while Viewing another Report
  3. Create a User Interface for Selecting and Launching Reports

- Provide "cleaner" response in situations where a user clicks "Cancel" from Dynamic Parameter dialogs.

Version 2.3.7: Released 12/17/2002

- This release addresses a rare problem with refreshing or loading the same report for the second time, when that report has both subreports as well as images. A problem in the Crystal runtime component was causing the viewer to fail in such cases.

- This version **allows you to view reports that result in zero records**.
  Thanks to Abbas Rostami (Cubic Corporation) for suggesting this functionality.

Version 2.3.5: Released 12/04/2002

- The Title Bar of the viewer now displays the report file name. For example,
  DataLink Viewer: My Report.rpt

- If the report contains a string formula named: **@DLV_Title** in the Report Header (even if that formula and/or section are suppressed), the Title Bar of the viewer will display the value of that formula.
  This allows you to override the default behavior of showing the report file name.
  It also allows you to build dynamic content for the Title Bar (using complex expressions to build up the string value.
  This functionality is demonstrated in the sample report:
  DataLink_Viewer_Year_and_Product_Prompts.rpt

- The ReportList.txt file, which holds the list of recently opened reports, no longer gets sent to the Recycle folder when it gets rebuilt (each time DataLink Viewer is closed).

  Thanks to Abbas Rostami (Cubic Corporation) for suggesting this functionality.

Version 2.2.4: Released 11/19/2002

- Dynamic parameter values are now automatically reused if the same Parameter is used more than once for the same report. For detail, see Sharing Values Across Cascading Parameters on page 83.

- DataLink Viewer now lets you launch another report by double-clicking a report row that contains a string formula with the following structure:

  // **invoke a report** (specify full path and report file name)
  'DLV_Run:-s "c:\directory\subdirectory\Report.rpt"

  *** or ***
  // invoke a report and specify **login information**:
  'DLV_Run:-s "Report.rpt" "user_id:dba" "password:sql"'

  *** or ***
  // invoke a report and specify (some or all) of the **parameter values**:
  'DLV_Run:-s "Report.rpt" "Parm1:1997" "Parm2:Gloves"'

  *** or ***
  // specify both **login as well as Parameter information**:
  'DLV_Run:-s "Report.rpt" "user_id:dba" "password:sql" "Parm1:1997"'

- You can invoke DataLink Viewer using a RUN command from Windows or from other programs. The structure is exactly as specified above, without the “**DLV_RUN:**” portion.
  For example, you can invoke viewing of the sample report while passing 1997 as the 1st parameter value, by issuing the following RUN command:

  "C:\Program Files\DataLink Viewer 9**DataLink_Viewer_9.exe**" -s "C:\Program Files\DataLink Viewer 9**DataLink_Viewer_Year_and_Product_Prompts.rpt**" "Parm1:1997"

- DataLink Viewer now lets you launch any other application and pass parameters to it by double-clicking a report section that contains a string formula with the following structure:

  'EXE_Run:Any_Program.exe:command line arguments'

Version 2.1.0: Released 11/01/2002

- **Corrected a problem with Using Report against Secure Databases.**
  The new version supports connection to any number of secure data sources by both the main report as well as the parameter reports.

  Thanks to Abbas Rostami (Cubic Corporation) for identifying the issue.

- **The title of the dynamic parameter windows now shows the text of the prompt designed for the parameter within Crystal**.

Version 2.0.0: Released 10/10/2002

- **Added compatibility with Crystal Report version 9.0.**
- Modified Print button to allow selection of printers.
- Changed cursor to hourglass while opening or refreshing reports.
- Enabled Export button in the viewer.
- Fixed a Refresh button issue

Version 1.2.0: Released 9/29/2002

- The Status Bar now shows the number of records **selected** in the report instead of number of records **read** from the database.

Version 1.1.0: Released 9/20/2002

- The application now accepts a report as a command line argument, allowing immediate preview of a report without the need to select it.
- You can now associate report file (*.rpt) with DataLink Viewer allowing users to directly open and preview a report by simply double-clicking it (as a file under a directory or an attachment in an e-mail message)
  Thanks to Michael O'Neill (*Alternatives For Industry)* for the enhancement idea.

Version 1.0.0: Released 8/23/2002

Known Issues and Limitation

1. .rpz files can't be exported to MAPI in a single step. The user is prompted to first export to disk, and then manually email the resulting file. Regular .rpt files don't have that restriction.

2. Only live parameters remember values from last session.

3. Patrick Pettibon from *MatchMarketing* (www.matchmarketing.com) has identified a problem and a solution to image quality issue when Previewing or Exporting (PDF/RTF/Word) Crystal Reports (8.5 & 9.0) that have charts, pictures, subreports, or OLE objects.

When working within Crystal Reports, the problem and solution are described by a Crystal Decisions tech support document at:
http://support.crystaldecisions.com/library/kbase/articles/c2009714.asp

You basically need to set the value of two keys:
1. BitMapMagnifDenominator (value should be 1)
2. BitMapMagnifNumerator (value should be 4)

Under the registry folders of:

For Crystal 8.5:
HKEY_CURRENT_USER\SOFTWARE\SEAGATE SOFTWARE\CRYSTAL REPORTS\EXPORT

For Crystal 9:
HKEY_CURRENT_USER\SOFTWARE\CRYSTAL DECISIONS\9.0\CRYSTAL REPORTS\EXPORT

However, when running any 3rd-party tool (such as DataLink Viewer) using the Crystal runtime components, similar settings are also required under a different registry folder.

You can download a zip file with detailed instructions, Before/After images, and REG files (one for CR 9.0 and one for CR 8.5 runtime components) that can be imported into the registry in order to set the required keys from:
http://www.milletsoftware.com/improving_image_quality.zip

The usual warnings about editing the registry apply.
If you don't have any image quality issues I suggest you take no action.

4. The Select Expert dialog has several limitations:
   - You need to have fields in the detail section of the report (even if that section is suppressed) before the Select Expert can use these fields.
   - There's a problem with using date fields in the Select Expert:
     http://support.businessobjects.com/library/kbase/articles/c2012687.asp
   - The XI version of of DLV shows only the 1st letter if you Browse for field values.