

REVELATION

S O F T W A R E

A R E V E L A T I O N W H I T E P A P E R

Converting, Front-ending or Migrating MultiValue Applications and Data for Use with OpenInsight



Contents

CONTENTS	3
INTRODUCTION	4
ACCESSING THE DATA FILES	6
OPENINSIGHT DATA DICTIONARIES	8
Dictionary Properties	9
Converting your data dictionaries for use within OpenInsight	11
The U2 Dictionary Converter Wizard	11
Understanding the relationship of dictionaries and data for use with an OpenInsight application.	12
CHARACTER TO OPENINSIGHT (CTO)	13
Using OpenInsight's CTO	14
Going GUI	17
CONCLUSION	18
Further details	18

COPYRIGHT NOTICE

© 2006 Revelation Software Limited. All rights reserved.

No part of this publication may be reproduced by any means, be it transmitted, transcribed, photocopied, stored in a retrieval system, or translated into any language in any form, without the written permission of Revelation Software Limited.

TRADEMARK NOTICE

OpenInsight is a registered trademark of Revelation Technologies, Inc. Advanced Revelation is a registered trademark of Revelation Technologies, Inc. OpenInsight for Workgroups is a registered trademark of Revelation Technologies, Inc. Report Builder+ is a trademark of Revelation Technologies, Inc.

Microsoft, MS, MS-DOS, Windows, are registered trademarks of Microsoft Corporation in the USA and other countries. IBM is a registered trademark of International Business Machines Corporation. Intel is a registered trademark of Intel Corporation. Lotus, Lotus Notes and Notes are registered trademarks of Lotus Development Corporation. All other product names are trademarks or registered trademarks of their respective owners.

Introduction

This white paper, adapted from a series of articles originally written by Robert Catalano, Revelation Software US and published in International Spectrum Magazine, is written for any MultiValue developer who wants, or needs, to fully exploit the modern Windows environment.

It might be that an existing character based system simply needs a GUI (Graphical User Interface) front-end to the existing database, or it might be that the business needs to share data with disparate systems or remotely through internet, wireless and other thin-client environments. Through OpenInsight, all of these needs and many more can easily and quickly be achieved.

Many of the common questions that are asked of Revelation Software by non-Revelation MultiValue developers will be answered over the following pages. Questions such as: "What is involved in converting another MultiValue application to OpenInsight?", "How do I utilize OpenInsight as a graphical front-end to my existing MultiValue application?" or "How can I migrate my data for use within an existing OpenInsight application?"

The path that you ultimately take will depend greatly upon what you want to do with your existing application. Usually there are four possible paths to take:

- 👁 A complete conversion from your current MultiValue system
- 👁 A gradual migration to OpenInsight
- 👁 A new front-end that utilises the existing database
- 👁 Conversion of your existing data for use with an OpenInsight application

Based upon your decision you will need to perform the following steps in some manner:

- 👁 Access the data files
- 👁 Review the existing application
- 👁 Set up the OpenInsight application
- 👁 Convert the dictionaries

This white paper has been divided up into three main sections. The first of these will focus on how you can simply provide access to your existing data files. In section two we will then go on to explore the data dictionary structure within OpenInsight, the conversion of the data dictionaries from traditional MultiValue systems to OpenInsight and the relationship of dictionaries and data for use within an OpenInsight application.

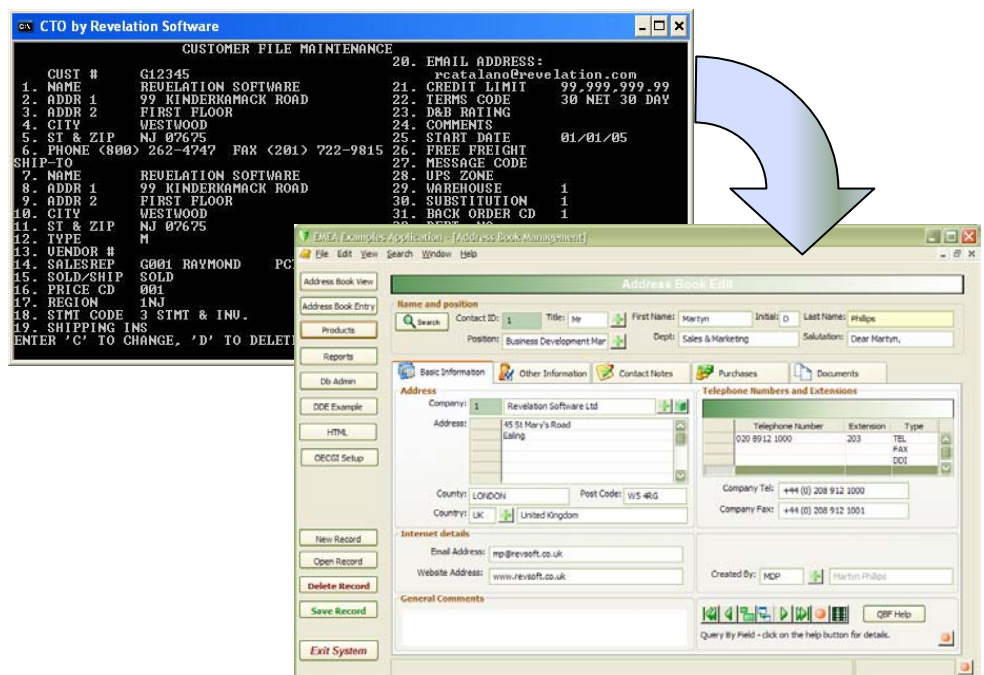
In the last of the three sections, we will look at the process of converting an existing MultiValue application called MYAPP to OpenInsight using OpenInsight’s CTO interface as well as using the Forms Designer to create a graphical interface to that application.

At Revelation Software, product development is an ongoing, continuous process and often enhancements are built into the product to make life easier for developers and users alike. One such enhancement is the inclusion of the Character to OpenInsight (CTO) interface.

With this new set of features, built into OpenInsight release 7.2 and above, MultiValue developers will be able to take an account-save of their existing application and restore the saved account as an application within OpenInsight. All files will be created in a specified location, all dictionaries will be converted as best as possible and character-based programs will be able to compile and run “as is” under a VT100 emulation.

This white paper aims to provide you with various options to extend the life of your legacy MultiValue systems and at the same time give them a future without having to dig deep into your IT budget, or erode much of this year’s profit line.

At the same time it is designed to provoke application migration specific questions and Revelation Software staff will readily welcome the opportunity to address those questions with you when you are ready.



Accessing the Data Files

If you want to copy your existing MultiValue data into OpenInsight tables there are several third party tools available to make this a simple process. The good news is that OpenInsight linear hash data tables are 100% compatible with your existing MultiValue files. Terminal emulators such as ViaDuct from Via Systems (www.via.com) and Accuterm from AccuSoft (www.accusoft.com) each have file transfer capabilities that can be used to copy your existing files to your workstation. Both of these terminal emulation products work on almost every version of MultiValue database.

For example, ViaDuct's 'Copy From Host' feature will allow you to save any file preserving the record, value and sub-value marks. Files created using this process are saved on your workstation with a .CFH extension. Once saved, the next step is to create a new table in OpenInsight (which is similar to CREATE-FILE) and write a simple script in Basic+ using the OSREAD command. This script will read the .CFH file and write it to your OpenInsight table.

Another useful tool for moving data into OpenInsight tables is the Zeus Data Integration tool from Mount Olympus Systems, Inc. (www.MtOlympus.us). This tool provides extensive mapping capabilities with an easy to use interface. In fact, OpenInsight 7.1.1 onwards includes a copy of the Zeus Inbound Engine, which works with many types of files, including .CSV and other delimited files, report images, UniVision XFH files, Universe Type 19 files and UniData Dir files. Figure 2 is a screen shot of the Zeus Data Integration tool transferring a Universe Type 19 file into an OpenInsight table.

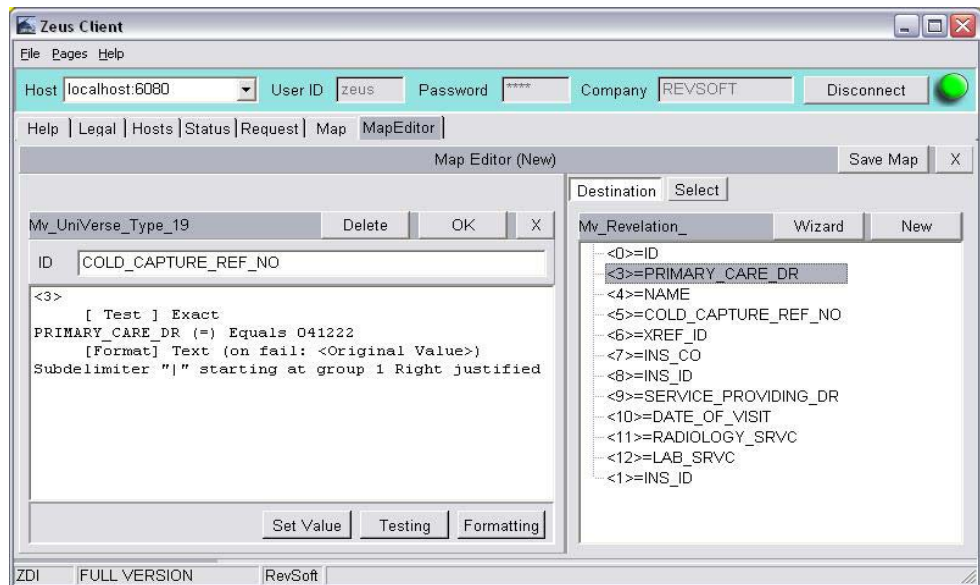


Figure 2: The Zeus Data Integration tool

If your goal is a gradual migration to OpenInsight or a new front-end to your existing U2 database then using an instance of a U2 BFS (Base File System) from within OpenInsight is the way to access your data. The first step is to create a connection to your U2 system using OpenInsight's built-in U2 Connector form.

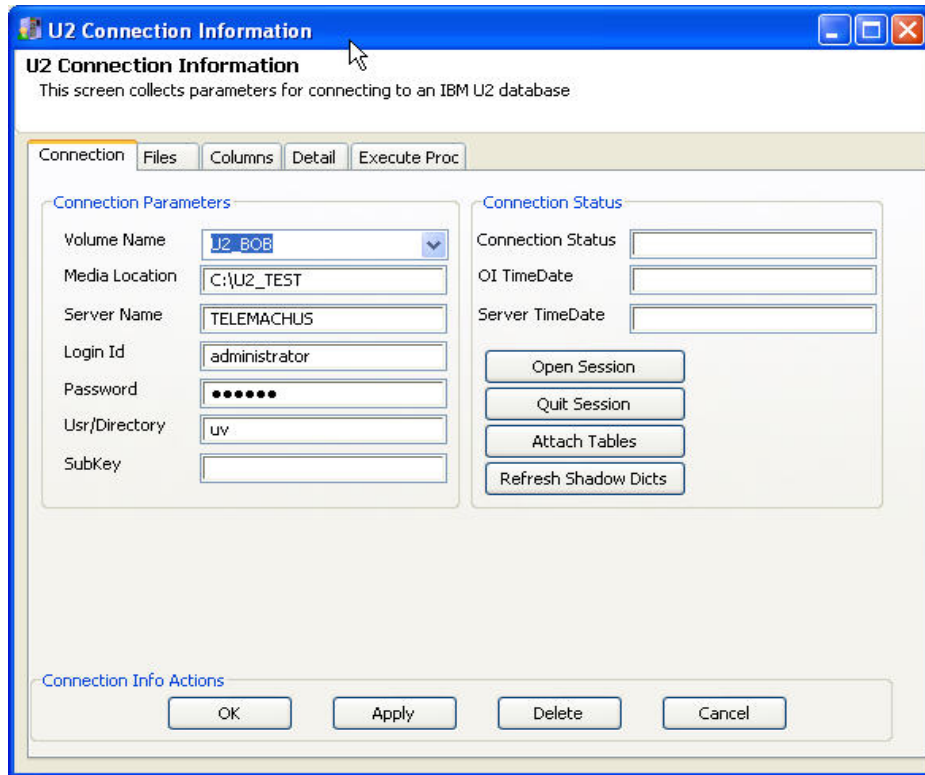


Figure 3: OpenInsight's built-in U2 Connector

Once a connection has been made, U2 files are attached as OpenInsight tables and are seen natively. A set of "shadow" dictionaries are converted from the U2 environment and stored within OpenInsight. Tools such as our Form Designer and Report Builder use the shadow dictionaries to point to the data stored within the U2 filing system. Any U2 file made available to the BFS can have a read, write or delete with record locking performed from within OpenInsight.

NOTE: Using OpenInsight as a GUI Front End to U2 is detailed in a separate paper available from Revelation Software entitled "OpenInsight as a GUI Front End to U2".

OpenInsight Data Dictionaries

During this section we will explore the data dictionary structure within OpenInsight, the conversion of data dictionaries from traditional MultiValue database technologies to OpenInsight and the relationship of dictionaries and data for use within an OpenInsight based application.

Each OpenInsight data table usually has a data dictionary associated with it, although this is not a definite requirement. The data dictionary stores column definitions for the data table. Each column you define is one entry (row) in the data dictionary table. A data dictionary is identified as DICT.TABLE_NAME where TABLE_NAME is the name you give the table; for example, if you name a table CUSTOMER, the associated dictionary is automatically named DICT.CUSTOMER.

Dictionaries are used any time that an OpenInsight tool accesses data, or when you want to access the data by column name; for example, when using TCL or the Report Builder.

If the data in a table is not structured and has no cross-row relationships, you are not required to have a dictionary; for example, a table containing text such as program code. In those cases, if you have a dictionary, there are no column definitions contained within it because dictionary column definitions are optional.

Once a table has been created, or when working with an existing table, data columns may be created by using the Table Builder tool. When you select a file in the Table Builder, the following window is displayed:

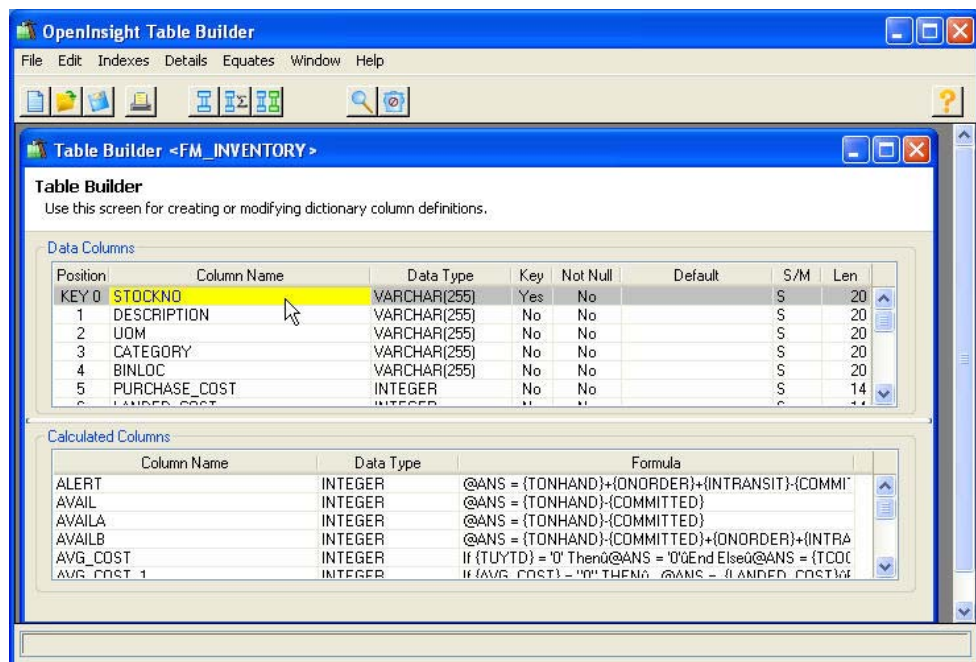


Figure 4: OpenInsight’s Table Builder

The dictionary components, as shown in the screen shot on the previous page, are as follows:

- 👁 **Position** is filled in automatically by the system and defines the position of the data column within the dictionary.
- 👁 **Column name** is the name of the field (known as columns in the OpenInsight world). Column names:
 - Can have a maximum of 50 characters and must begin with an alpha character.
 - Can use the following legal characters: upper case A-Z, 0-9, and _ (underscore).
 - Cannot be a system-reserved word, such as SELECT, WITH, SORT, etc.
 - Must be unique within the data dictionary.
- 👁 **Data Type** is the data type of the column; for example BOOLEAN, CHAR, DATE, INTEGER, VARCHAR.
- 👁 **Key** is a Boolean Field indicating whether or not the column is the key for the table.
- 👁 **Not Null** indicates whether or not data must be entered into this field before the record is saved.
- 👁 **Default** is the data that will appear as a default.
- 👁 **S/M** is the value type, where S is SingleValue and M is MultiValue.
- 👁 **Len** is the display length of the field.

Dictionary Properties

OpenInsight allows you to define calculated (or “symbolic”) columns, giving you the full power of Basic+ in your dictionaries. These are used in place of the correlatives found in traditional MultiValue dictionaries.

The formulas that define calculated (or symbolic) columns are stored in the data dictionary. When you build a formula, you may use the names of existing columns to define the formula from which the calculated column is derived.

For example, T correlatives are replaced using the XLATE function, an example of which is copied below and as shown in the OpenInsight Table Builder ‘Edit Data Column Attributes’ window on the next page:

```
@ANS = XLATE("FM_CATEGORY", {CATEGORY}, "DESCRIPTION", "X")
```

There are a number of developer definable Data Column attributes and many are common to normal columns and calculated columns. As we are currently talking about the use of calculated columns, that attribute sheet is shown and described below. Our OpenInsight tutorials designed for use with the free evaluation software (<http://www.revsoft.co.uk/tutorial.pdf>) explore OpenInsight’s screens in more detail.

- 👁️ **Name** is the name of the calculated column.
- 👁️ **Data Type** is the data type for this column.
- 👁️ **Output** is the Output Conversion or OCONV for this column.
- 👁️ **Multivalued**, when checked, indicates that multiple data values may be stored in this column.
- 👁️ **Left/Center/Right/Text** is a radio button group, which indicates how the data will be displayed when printed on reports.
- 👁️ **Length** is the default display length. It does not limit the amount of data that can be stored in this column.
- 👁️ **Column Heading** (For Reports) is the default column heading that will appear in reports, or when this column is added to a data entry form using the Form Designer tool.
- 👁️ **Edit Formula** is the formula for the calculated column. This formula is used to determine the values for the column. In the illustration above, the result (returned in @ANS) is the product category description.
- 👁️ The **Test button** is used to check the syntax of the calculated column.
- 👁️ **Description of Column Usage** is a text field used for documentation purposes.

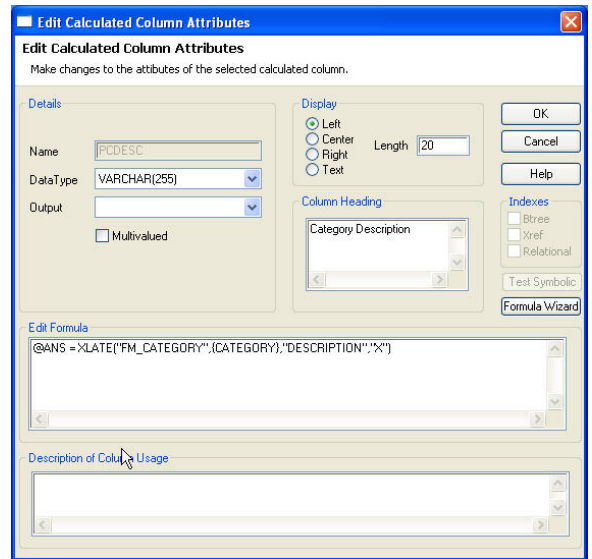


Figure 5: Edit Calculated Column Attributes form

There are many uses for calculated columns. For example, an INVOICE_TOTAL column might contain a formula that multiplies the value in the QUANTITY column by the value in the PRICE column. The calculated field formula could also be a function to look up values and extract data from a related table, or perform some other processing.

Converting your data dictionaries for use within OpenInsight

OpenInsight includes a “U2 Dictionary Converter” tool to automate the process of converting your existing data dictionaries. In section one of this paper, we discussed the various ways you could import your dictionaries and data into an OpenInsight table. The “U2 Dictionary Converter” tool is designed to convert and move the existing dictionaries to their proper location.

The U2 Dictionary Converter Wizard

OpenInsight’s built-in U2 dictionary conversion program, as depicted in section one, can be utilised to save your traditional dictionaries into a table called DICT_INVENTORY.

The dictionary conversion wizard can then be used to convert and automatically move the data dictionaries from the table DICT_INVENTORY to the table DICT.INVENTORY for you.

Once the dictionaries are converted they can be edited as standard OpenInsight dictionaries by simply using the Table Builder tool as shown in figure 6.

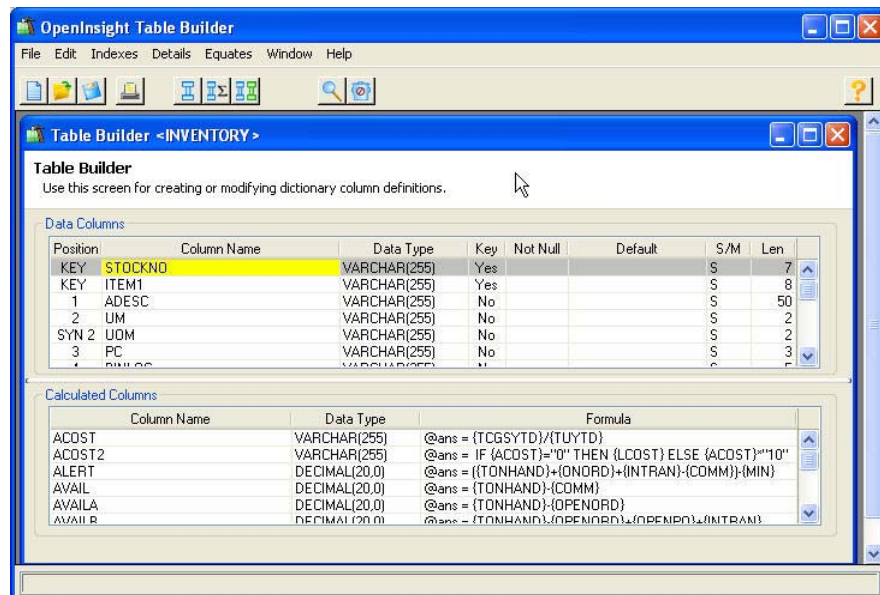
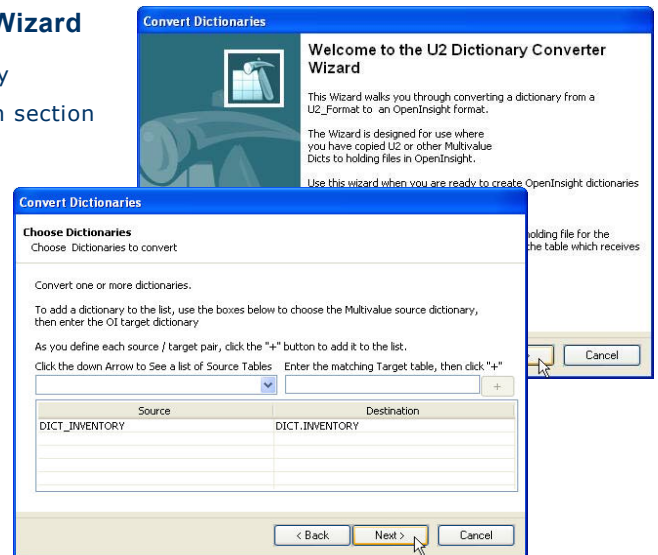


Figure 6: OpenInsight’s Table Builder

Understanding the relationship of dictionaries and data for use with an OpenInsight application.

In traditional MultiValue systems developers create an account and then create files (or tables) within that account. The programs, dictionaries and data all exist under that account name (with the exception of Q pointers). In OpenInsight, developers create an application and also create tables, specify the location of where the tables exist, and then attach those tables for use within the application. In fact, both the dictionaries and data may exist anywhere on the network.

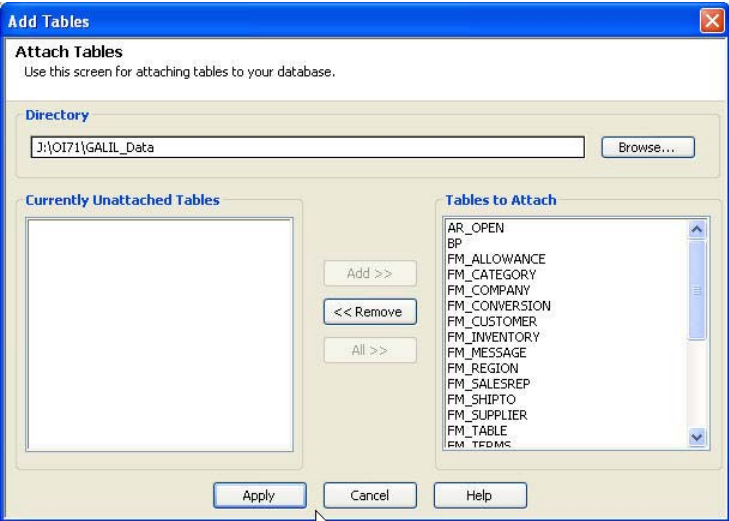


Figure 7: OpenInsight’s Attach Tables form.

Once your data dictionaries are converted and attached to your application you are now ready to begin the process of creating forms, reports and programs for use with your data.

Character to OpenInsight (CTO)

CTO is a new set of features that are built into OpenInsight version 7.2 and above and enable MultiValue developers to take a virtual ACCOUNT-SAVE of their existing application and restore the saved account as an application within OpenInsight. All files will be created in a specified location, all dictionaries will be converted as best as possible and character-based programs will be able to be pre-compiled and run 'as is' under a VT100 emulation.

The CTO interface supplements the GUI interface that is synonymous with OpenInsight. Both the GUI and CTO interfaces communicate with the Revelation OpenEngine (the heart of the database environment). Therefore both interfaces have access to all of the files and programs stored in the database and they can interoperate.

Developers can have the best of both worlds (GUI and CTO), with no middleware add-on costs, as well as take advantage of OpenInsight's economical Server Deployment Packs.

Rather than drawing Windows or Linux Desktop graphical forms however, the CTO handles the processing of traditional MultiValue PRINT, INPUT, etc., statements, including support for cursor movement and screen attribute control.

In addition, CTO emulates a toolset and environment that are familiar to the traditional MultiValue developer and user, including a command line prompt, support for PROCs and Paragraphs, a 'master dictionary', a spooler interface and system tools like ED, BASIC, LIST/SORT, etc.

Once the CTO conversion is complete, the MultiValue developer is provided with a number of choices. With OpenInsight's CTO interface, developers can run existing character-based applications 'as is' and migrate them to a graphical user interface on an 'as needed' basis. Alternatively, a hybrid application can be created, made up of a mixture of character and graphical user interface forms all within the same application.

CTO is included in OpenInsight at no additional charge. With OpenInsight release 7.2 and above, MultiValue developers can create professional looking, scalable, extensible, robust applications in a fraction of the time it takes to create them using other development tools, and deploy single user runtime versions at no extra cost. A 250-user system costs less than £20.00 per seat with no annual maintenance fees.

Using OpenInsight's CTO

For the remainder of this section we will work on converting a standard MultiValue application called MYAPP.

The CTO interface can be invoked from a terminal emulator using telnet, or directly from the Windows command prompt. Figure 8 is a screen shot of the OpenInsight character interface logged in to the SYSPROG application.

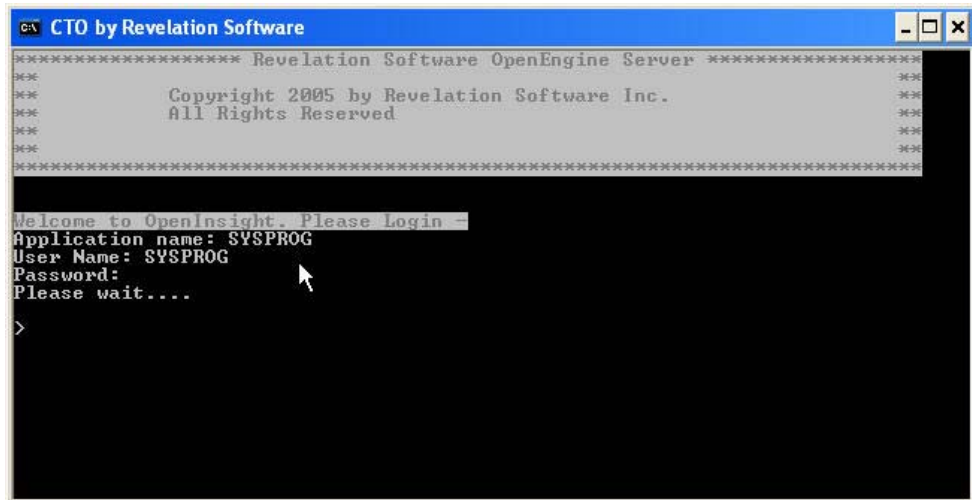


Figure 8

As discussed previously, data is stored externally from the application and may exist anywhere on the network. For our sample conversion we will create a folder called MYApp_Data located on our server. The dictionaries and data from our virtual ACCOUNT-SAVE will be restored to this folder. Utilising the line editor provided by the CTO interface as shown in Figure 9 we create a conversion record called MYAPP*TRANSLATE in the master dictionary table (MD).

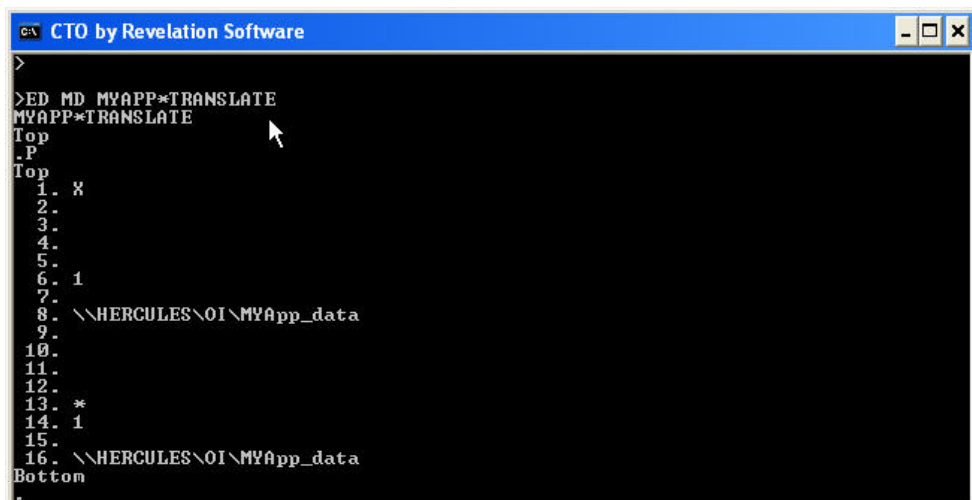


Figure 9

Next, as shown in Figure 10, we will select our virtual tape and proceed with the ACCOUNT-RESTORE. The CTO interface supports the MultiValue terminal control language (TCL).

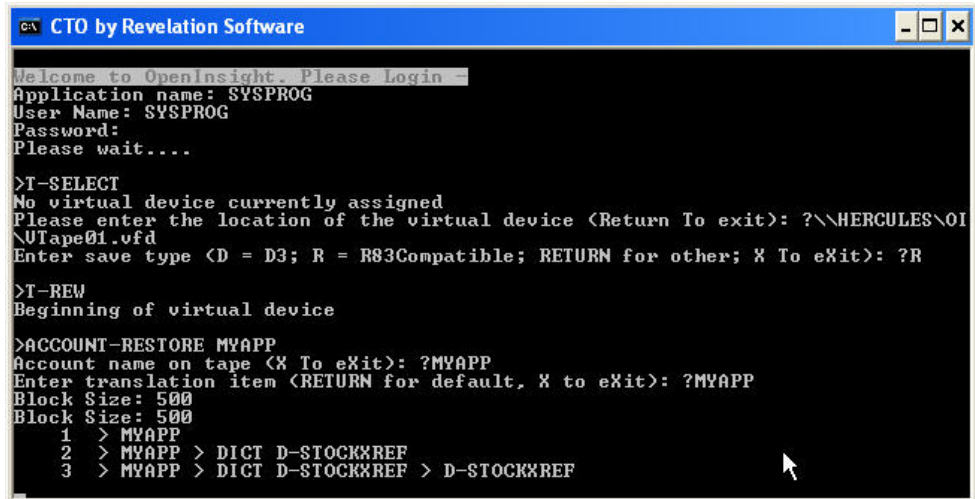


Figure 10

After the restore is completed our MYAPP application has been created within OpenInsight. Our dictionaries and data have been converted to OpenInsight and are now ready for use with our Form Designer and Report Builder tools. Figure 11 depicts how the ACCOUNT-RESTORE process automatically converted correlatives and conversions to OpenInsight calculated columns.

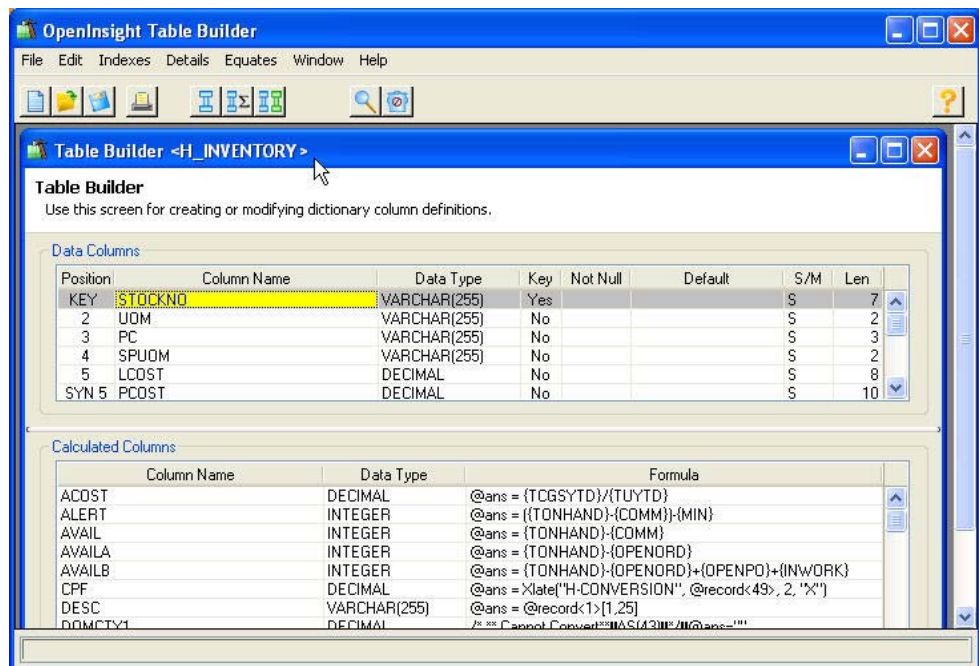


Figure 11

To complete our conversion process we now need to compile our BASIC programs so that they can run within OpenInsight as a character application. The source code from our original application is passed through a pre-compiler and then compiled for use within OpenInsight. As mentioned before, the OpenInsight CTO interface also provides support for PROCs and Paragraphs.

```

c:\ CTO by Revelation Software
-----
MV APPLICATION CORPORATION
M A I N   S E L E C T O R
01:19:03PM

1. Order Processing
2. Invoicing
3. Accounts Receivable
4. Purchase Order Processing
5. Pricing/Costing/Quotations
6. Inventory
7. Sales Analysis
8. File Maintenance
9. Accounts Payable
10. General Ledger
11. End Of Period Processing
12. Printer Functions
13. Reset Laser Printer Settings
14. Inquiry
15. Warehousing

ENTER SUB-SYSTEM CODE, 'END' OR 'OFF' TO LOG-OFF :-
    
```

Figure 12 – Example of PROC output

```

c:\ CTO by Revelation Software
-----
CUSTOMER FILE MAINTENANCE
-----
CUST #      G12345
1. NAME     REVELATION SOFTWARE
2. ADDR 1   99 KINDERRAMACK ROAD
3. ADDR 2   FIRST FLOOR
4. CITY     WESTWOOD
5. ST & ZIP  NJ 07675
6. PHONE (800) 262-4747 FAX (201) 722-9815
SHIP-TO
7. NAME     REVELATION SOFTWARE
8. ADDR 1   99 KINDERRAMACK ROAD
9. ADDR 2   FIRST FLOOR
10. CITY    WESTWOOD
11. ST & ZIP NJ 07675
12. TYPE    M
13. UENDOR #
14. SALESREP G001 RAYMOND PCT= 010
15. SOLD/SHIP SOLD
16. PRICE CD 001
17. REGION  1NJ
18. STMT CODE 3 STMT & INU.
19. SHIPPING INS
20. EMAIL ADDRESS: rcatalano@revelation.com
21. CREDIT LIMIT 99,999,999.99
22. TERMS CODE 30 NET 30 DAY
23. D&B RATING
24. COMMENTS
25. START DATE 01/01/05
26. FREE FREIGHT
27. MESSAGE CODE
28. UPS ZONE
29. WAREHOUSE 1
30. SUBSTITUTION 1
31. BACK ORDER CD 1
32. DEPT. NO.
33. HISTORY FLAG Y
34. BOOKING LIMIT .00
35. SHIP VIA UPS
36. CONTACT 1 MIKE RUANE
37. CONTACT 2 ROBERT CATALANO
38. CONTACT 3
39. A/R ACCT# G12345

ENTER 'C' TO CHANGE, 'D' TO DELETE, 'P' TO PAGE OR 'E' TO END - >
    
```

Figure 13 – Example of BASIC output

Going GUI

Now that our application has been converted “as is” to OpenInsight, the next step is to convert the existing program to a graphical format. This is accomplished using the Form Designer tool within OpenInsight. This tool creates graphical forms using the data dictionaries that were converted during the ACCOUNT-RESTORE process.

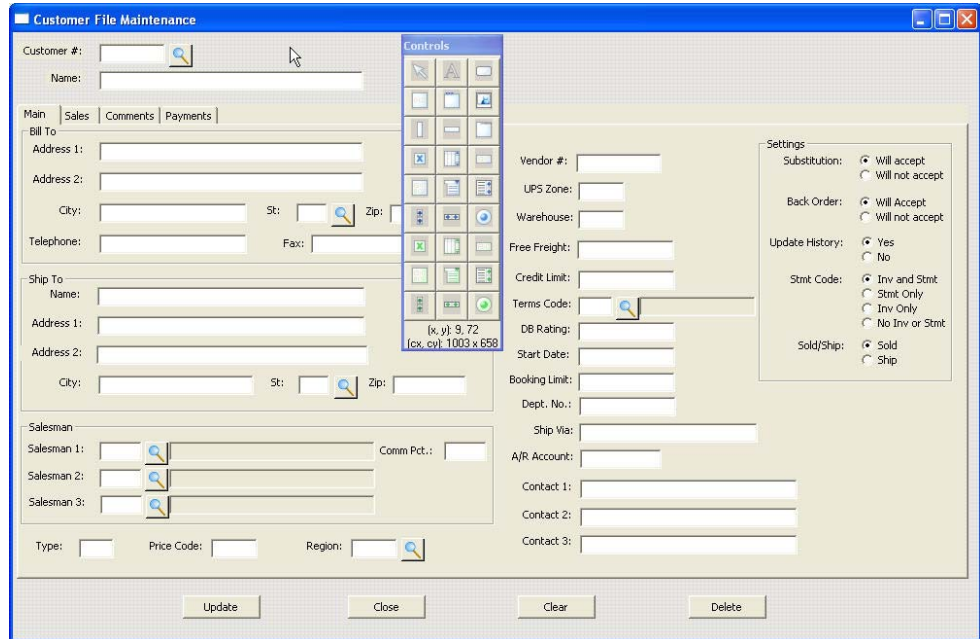


Figure 7 – Creating a form using the Form Designer tool

You can also use the various other GUI development tools to quickly and easily create dynamic pop-ups, messages, help, menu structures, context help and reports as well as advanced functionality such as web forms, XML functionality, Data Warehouses, and much more.

Conclusion

Now, the choice is yours: with OpenInsight's CTO interface you can run your existing character based application "as is" and migrate to a graphical interface on an "as needed" basis. In fact, you can have a mixture of character and graphical forms all within the same application.

In the first section of this paper, we posed the following questions: "What is involved in converting another MultiValue application to OpenInsight?", "How do I utilise OpenInsight as a graphical front-end to my existing MultiValue application?" and "How can I migrate my data for use within an existing OpenInsight application?"

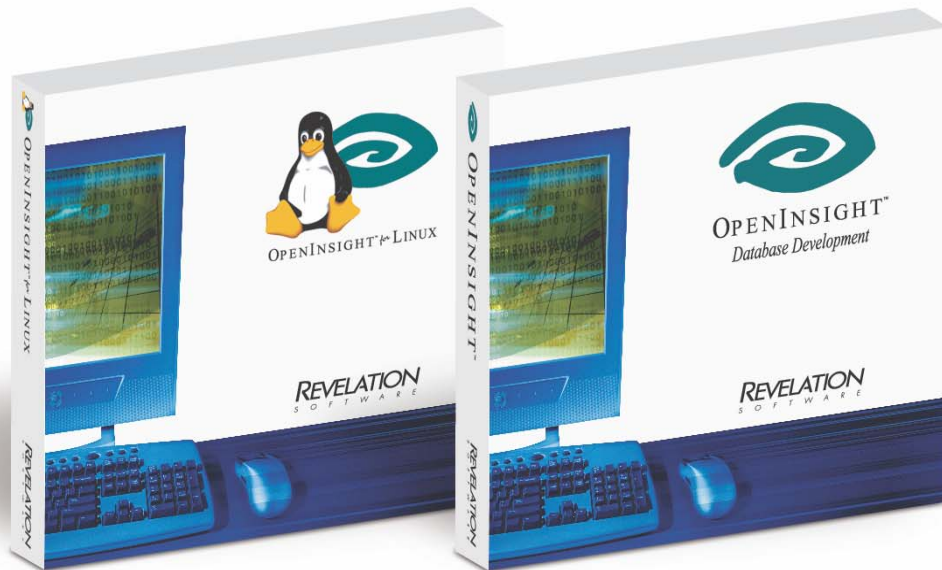
Further details

We hope that this paper has answered these questions and demonstrated a clear path to converting, migrating or front-ending your existing application for use within OpenInsight. If you have any questions, please feel free to contact your nearest Revelation office, the details for which can be found on the very last page.

In addition an evaluation version of OpenInsight can be downloaded from our web sites giving you the opportunity to road test OpenInsight for a full 30 days prior to making any purchase decision. The web sites also include several documents to help you take your first steps in further developing fully featured GUI solutions with OpenInsight.

Some useful links are:

-  <http://www.revsoft.co.uk/oievalmain.htm>
Access to all of the versions of evaluation software for use with Windows and several flavours of Linux.
-  <http://www.revsoft.co.uk/oievalmain.htm>
A comprehensive step by step tutorial that takes a look at all of the basic concepts of OpenInsight from table and form design, through pop-ups and messages, menus and web enabling your application.
-  <http://www.revsoft.co.uk/Downloads/eval/oi71quickstart.pdf>
A quick start guide to working with the evaluation software, designed to get you going with OpenInsight very quickly.
-  <http://www.revsoft.co.uk/downloads.htm#wp>
A number of useful developer resources including white papers on topics such as Using MS Excels Web Query functionality with OpenInsight, Using XML with OpenInsight and Creating Multi Table Windows and more.



OpenInsight for Workgroups - a repository-based applications development environment that enables development teams to work collaboratively to design, develop, deploy and scale high-performance business solutions. OpenInsight's complete tool set (including a Form Designer, Editor/Debugger, UI Workspace, XML Workspace and more), advanced BASIC+ scripting language and powerful client/server development combine to deliver a complete environment for deploying and evolving network applications. OpenInsight is delivered with a flexible Multi-Valued, variable length, filing system, or has the ability to use SQL Server, Oracle, Lotus Notes, Universe/UniData, D3, or any ODBC compliant data back-end. Delivered with both XML and Web-enabled tools, OpenInsight is a tremendously powerful development tool for Windows and Linux-based applications.

JOI - an integrated environment for managing the entire lifecycle of Java component based enterprise applications. JOI provides a single centre and integrated toolset for automating the design, development, assembly, deployment, customization, extension, and maintenance of Java component-based enterprise applications, tailored to the specific needs of the Revelation developer.

Revelation Software – Delivering feature rich MultiValue application development tools that empower both developers and users alike.

Experience the toolset at www.revsoft.co.uk

REVELATION

S O F T W A R E

With more than 1.6 million users, at over 80,000 sites, on all seven continents, Revelation Software is in nearly every industry imaginable.

Revelation Software specialise in the provision of database development tools that permit applications to be developed in a fraction of the time required by traditional development environments. Revelation products may be scaled from single-user to large enterprise wide networks, from the desktop to the World Wide Web - for a fraction of the cost of other enterprise software solutions.

Revelation has a proud heritage, built on the provision of PC Database Software since 1982. During this time, Revelation has embraced the benefits of "post-relational" data structures that are now being more readily embraced by the mainstream.

Revelation applications are characterized by the enduring nature of the solutions implemented using its products. Revelation continues to invest in technologies that enable systems implemented in its earliest development tools to operate successfully on modern 32-bit client and network operating systems, and be extended to the Web - 20 years on, Revelation-based systems are still working hard for their owners.

Revelation continues to provide new tools that natively exploit the capabilities of the latest operating systems, network computing platforms, and programming languages and techniques. The combination of this focus and a commitment to continuous improvement and innovation has enabled Revelation to thrive and prosper in the challenging and ever-changing market of application development tools and deployment technologies.

"Social Software Limited has relied on Revelation for over 18 years. Throughout numerous changes in operating environments as well as unexpected changes in application requirements there has been no substitute tool which has offered the same development and maintenance benefits.

Thus we have developed an extremely mature software library including reporting tools which significantly extend the Revelation's environment in order to achieve a tight fit with our marketplace. As application developers we have complete confidence that Revelation will be a key part of our capability to satisfy any kind of application demand and challenge which typically exceeds customer expectations in our chosen domains."

Marc Radley, Managing Director – Social Software Limited

Revelation Software USA

99 Kinderkamack Road
Suite 109
Westwood, NJ 07675
United States of America

Toll Free: (800) 262-4747
Phone: (201) 594-1422
Fax: (201) 722-9815
Email: info@revelation.com

Revelation Software UK

45 St Marys Road
Ealing
London, W5 5RG
United Kingdom

Phone: +44 (0) 20 8912 1000
Fax: +44 (0) 20 8912 1001
Email: info@revsoft.co.uk

Revelation Software Australia

Unit 2
21 Old Pitwatter Rd.
P.O. Box 300
Brookvale, NSW 2100
Australia

Phone: +61-2-9939-6399
Fax: +61-2-9939-6366
Email: info@revelationsoftware.com.au

- www.revsoft.co.uk -