

INTERNATIONAL Spectrum

THE MULTIVALUE  TECHNOLOGY MAGAZINE | SEPTEMBER/OCTOBER 2015

Reviewing

the Idea of

Code Review

ALSO IN THIS ISSUE

- Finding Your Inner API Designer.
- REST is not Rest. Here's the Difference.
- This Issue: Business Tech is All About Hate.
- You're Productive. Look the Part!

April

11 - 14

2016

The Premier MultiValue User Event



From application development and technical systems support to complex business and web development, we offer complete IT solutions.

RETURNING TO THE WIGWAM , LITCHFIELD PARK, ARIZONA



The **International Spectrum MultiValue Conference** is the only place that provides diverse solutions and support for integrating and developing your Enterprise applications into the world of business and technology.

This annual event is the place to find the **knowledge, people, resources,** and **technologies** for enhancing your MultiValue (D3, UniVerse, UniData, Reality, jBase, Caché, QM, and OpenInsight) software applications.

Topics covered at the International Spectrum MultiValue Conference include:

- **Integration**
- **Database Management**
- **Mobile Integration**
- **Reporting and Data Presentations**
- **Project Management**

...and more.

REGISTER AT

www.intl-spectrum.com/conference





6 Code Review Code review is a hot-button topic for many programmers. You probably already know the arguments against. Susan Joslyn takes up the cause. Let her inspire you to make it part of your project lifecycle. **BY SUSAN JOSLYN**

FEATURES | SEPTEMBER/OCTOBER 2015

9 Perspective Shift: Thinking in APIs Building Monolithic programs makes sense in a stateful, telnet-based world. The web is a more API-centric place. Kevin invites us to join him in some creative perspective shifting. **BY KEVIN KING**

15 OpenInsight Data Encryption at REST RESTful encryption is a key part of modern application security. Get some insight into how Revelation makes it work for your programs. **BY ROBERT CATALANO, DIRECTOR OF SALES, REVELATION SOFTWARE**

19 Business Tech: Why I Hate Technology Ever been blamed by a person in your company for the hardware and software decisions that happened before you got there? Been rewarded with a smile because they got a new computer or monitor, even if you had nothing to do with it? This article is about that. **BY CHARLES BAROUCH**

23 Clif Notes: New Blood Part 4 If green-screens look out-of-date to your boss, people who spend their day on green-screens might start looking out-of-date, too. Learning new tools doesn't just make you more productive, it makes you *look* more productive. **BY CLIFTON OLIVER**

DEPARTMENTS

From the Inside page 4

From the Press Room page 12

International Spectrum and MultiValue are registered trademarks of International Spectrum, Inc. All other registered trademarks are the property of the respective trademark holders.

From the Inside

How do you build a new business application using all the current technologies and design patterns? I've been thinking about this for a while now. So, I've decided to take on the task of reinventing the wheel.

Why do this? There are tons of applications out there that already do accounting, inventory, manufacturing, and/or service related offerings. Why put any effort into recreating all that work?

My main reason is because it feels like no one has really tried recently. There are a lot of "bolt-on" applications that are designed to work with existing LOB (Line-of-Business) software and existing accounting applications. I've even seen some open source applications out there that tried to tackle this sort of fresh approach, but they are designed narrowly, to solve a specific interface task more so than to reap the advantages of current technology and design practices. These applications work well for what they are, but stop short of what businesses will need to meet future requirements.

So, for better or worse, I'm starting an article series that goes through the practice of building an accounting application using the "API" or "Micro Services" design approach.

Since most of you already have an existing, well used, and hopefully well loved, LOB application running your company, I hope you will be inspired by these articles, sessions, and code samples that will show different ways to solve the problems associated with

building the next generation of LOB applications.

Spectrum published a series of articles on Inventory Management <<http://intl-spectrum.com/mag/MARAPR.2006/default.aspx>> some years back. Those articles asked us to re-examine that part of the business. This series will do something similar for accounting.

So for starters, what are some of the things the next generation of LOB software must have?

API and Micro Services

APIs are simple, modular routines that are designed to access specific business processes. Sometimes, APIs are just data access, but many times the APIs are accessing *other* APIs to do more complex tasks. We already do this, often without realizing it, when creating Multi-Value BASIC subroutines.

So, APIs are those individual "access/end points" into an application that do specific functions like: Post GL, Reverse GL, Update Inventory Running Totals, Read Customer Information, and so on. Micro Services are a new concept that puts a *shell* around the API application development, creating an interchangeable application structure. I'll get into Micro Services and their designs later in the series.

Adaptive Interfaces for Users

Every user is different, and every user interfaces with the company in different ways. I'm not saying that every API has to be available from every interface, but a failure to adapt to different interfaces is a shortcoming of many LOB appli-

INTERNATIONAL Spectrum

SEPTEMBER/OCTOBER 2015

NATHAN RECTOR
President

CHARLES BAROUCH
Editor

SYDNEY BAROUCH
Editor

TRACEY RECTOR
Layout



Learn more about the MultiValue Symbol and see what MultiValue Technologies and MultiValue Communities exist to help you support and manage your business and systems. To find out more visit

<http://www.intl-spectrum.com>

MISSION STATEMENT *International*

Spectrum magazine's editorial mission is to be the premier independent source of useful information for users, developers, and resellers of MultiValue database management systems, open systems business database solutions, and related hardware, software, and peripherals. Published bimonthly, *International Spectrum* provides comprehensive coverage of the products, companies, and trends that shape the MultiValue marketplace as well as the computer industry at large — helping its readers get the most out of their business computer systems.

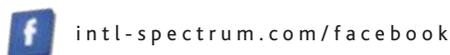
International Spectrum is published six (6) times per year at the subscription price of \$40.00 U.S. in the U.S.A.; \$45.00 U.S. in Canada and Mexico; \$50.00 U.S. for other countries. Single copy rates are \$7.00 U.S. in the U.S.A. and Canada, and \$9.00 U.S. in all other countries. *International Spectrum* is published by International Spectrum, Inc., 3691 E. 102nd Ct., Thornton, CO 80229; Tel: 720/259-1356; Fax: 603/250-0664 E-Mail: request@intl-spectrum.com. Copyright 2015 International Spectrum, Inc. All rights reserved. Reproduction in whole or in part, without written permission, is prohibited.

PRINTED IN USA

NEWS RELEASES/UNSOLICITED ARTICLES

International Spectrum is eager to print your submissions of up-to-the-minute news and feature stories complementary to the MultiValue marketplace. Black and white or color photographs and diagrams are welcome. Although there is no guarantee a submitted article will be published, every article will be considered. Please send your press releases, articles, and queries to: editor@intl-spectrum.com. *International Spectrum* retains all reprint rights.

International Spectrum is a registered trademark and MultiValue is a trademark of International Spectrum, Inc. All other registered trademarks and trademarks are the property of the respective trademark holders.



cations. Our new LOB APIs must be workable from any interface that the user has access to, within reason.

If the user wants to input data using Excel, then provide them with an interface that allows them to create, retrieve, and update from Excel. If they need mobile access, then provide a rich mobile interface. If you have a customer facing website, then provide real-time, secure interfaces to customer data. For the users that work at their desk all day, provide them a desktop application. If you have a user that just needs a dashboard, then provide them a dashboard.

You get the point. Allow the user to define how they interact with the API, never allow the API define what interfaces the user can access.

Reporting, Smart Data, and Business Intelligence

It doesn't do an application any good to store information but never do anything with it. Unfortunately, a lot of LOB applications do exactly

that. They provide some BI and some reporting, but nothing really good in the way of interfaces to create *Smart Data*.

Interfacing with *Data Lakes*, *Shared Data systems*, or *Big Data* is a must for any LOB, even if there isn't a pre-identified need. Restricting yourself to just looking at the data you've collected can become a business trap, so pulling in data from other locations to enhance the information you already have is a must. Even if it is something as simple as combining what you have with data drawn from Social Media.

Interfacing with Outside Systems

Okay, this may sound like a repeat of the two above, but it is really important in its own right. B2B sources aren't just useful for the data you can import. *Exchanging* information with the same B2B sources is equally as important. How many times have you had to interface with other systems through some kludgy methodology to get the simplest piece of information? EDI comes to mind.

There are tons of web services out there that can process or manipulate our data better than what we can write from scratch in a reasonable amount of time. This means exporting data and importing data is a must, even when the other system has no additional data to offer. Business applications are not just about the data, but also about the business logic for processing that data.

The New Series: Starting Next Issue

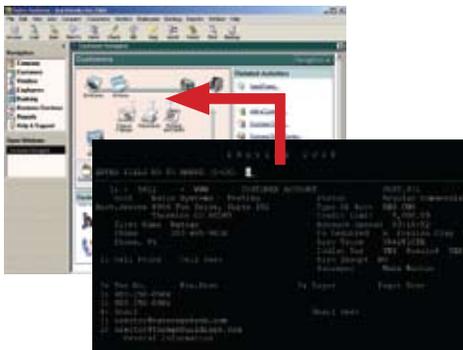
I know I'm taking on a substantial task. Whatever input or help that people are willing to provide would be welcome.



NATHAN RECTOR
President
International Spectrum
nathan@intl-spectrum.com

mv
QB

QuickBooks API for the MultiValue Database



- **Read/Write Directly to Quickbooks Databases**
Customer, Vendor, Invoices, Purchase Orders, Chart of Accounts
- **mvQB API is Designed for the MultiValue Program to Use**
All routines are simple BASIC calls designed for the developer. No special user interfaces required.
- **No Need to Learn the Internals of QuickBooks**
- **QuickBooks Pro/Premier/Enterprise**

NATEC
Systems



Providing Solutions to your MultiValue Questions

Phone: 303.465.9616
E-mail: mvqb@natecsystems.com
Website: www.natecsystems.com



Code Review

BY SUSAN JOSLYN

Writing software is a skill that can be acquired and honed with practice. It can indeed be artful, and some people have an uncanny talent for it, but it is still a science. And just as a good novel is the creative endeavor of its author, a solid grasp of language and adherence to syntax guidelines are what make it readable.

Today's programmer must be conscious of governance, security issues, the old-fashioned notions of quality, and the ultimate business question: *Is this software doing us some good?* Thus, a certain amount of our attention must focus on testing and review. Test a program to see what it does. Review it to see how it does it.

Often testing is performed in a 'black box' style, meaning that the person executing the software is purposefully not aware of the mechanics, just the resulting behavior. The downside of this? After fifty years we have all seen

To catch bugs. To insure that code is readable and maintainable – and fairly standard. To spread knowledge...

MultiValue [*and non-MultiValue – Ed*] solutions that work great, but when you take a peek under the hood, Italian food comes to mind. Spaghetti code. And who hasn't stepped in as a contractor or a new hire and had to slog through a bunch of poorly written and inconsistently maintained software? Code review can be the cure.

There is a difference between code review and peer review, but in many ways they are the same. For one thing, in either case you want to have someone else do the reviewing; not the person who did the coding. There is nothing like a fresh pair of eyes! Peer review may be more casual and often includes

a bit of unit testing, where code review, by its name alone, is about looking inside the box at the actual mechanics of the software: The source code.

In both cases it is convenient to have a checklist; and in both cases it is important to keep the checklist fairly short and flexible. Code review tends to be the more rigorous of the two processes. It is often performed by individuals who are dedicated and have, shall we say, the personality for it.

Jeff Atwood is an American software developer, author, blogger, and entrepreneur. He is known for the programming blog Coding Horror, and is the co-founder of the question-and-answer websites Stack Overflow and the Stack Exchange. Most of us know who he is; many of us respect his opinion. He recently voiced his view on this topic:

"Peer code reviews are the single biggest thing you can do to improve your code."

PICK PROFESSIONAL

Don't make the mistake of placing your career in the hands of just ANY search firm. We are professionals who understand the importance of CONFIDENTIALITY and RESPECT of a job search, and our database of clients is one of the largest in the country. Unlike the rest, we will work in YOUR best interests' to help you further your career. Because of our dedication and professionalism, we are recognized as the leaders in the PICK/UniVerse/Unidata placement industry in the Tri-State area and throughout the U.S. So if you are tired of putting yourself at the mercy of the rest.

CALL THE BEST! Contact...

Matt Hart

EXECU-SYS, LTD

1411 Broadway, Suite 1220

New York, NY 10018

(212) 967-0505

(800) 423-1964 x 302

Fax: (212)947-8593

Email: mh@eslhy.com

Consultants Needed Nationwide

What is the purpose of code reviewing? To catch bugs. To insure that code is readable and maintainable — and fairly standard. To spread knowledge of the code base throughout the team as well as to get new people up to speed. It is also great for exposing programmers to different approaches and often little nuggets of wisdom. How often have you looked over another programmer's shoulder and said "I didn't know you could do that!?"

If code review is so great, why doesn't every development group have a code review stage in their lifecycle? Everything boils down to the idea that *there simply isn't time*. That old chestnut about a stitch in time saving nine merely inspires a snorted "prove it!" We probably can't — not on a specific basis — until the practice is implemented and metrics are reviewed. Plus, programmers resent testers, and perhaps code

reviewers even more, because they (we) don't want another programmer — a peer, no less — to be looking over our work and possibly criticizing it. Yes, it's really just another pair of eyes helping you make sure your work reflects your best effort... but it feels like judgment. And hassle. We resent the idea of code reviewers having authority.

In many cases that resentment is justified. Who wants a petty tyrant running the code reviews? There are ways to relieve that pressure, of course. One way is to create a guideline or checklist, but make sure that it is not onerous. And be sure to allow it to evolve with feedback from all parties.

What might a code reviewing checklist include? The core imperative is to organize complexity. Source code should be clear and readable. The intention of the program should be revealed by looking at its source. Here are some starting points:

- Source code should be clean. One important way to achieve this is by deleting — not commenting out, *deleting* — obsolete code. There may be a valid reason for keeping a remarked section but it would usually be temporary. It should not be standard practice. Your source code control software will manage the change (and if it doesn't, give us a call).
- There should be no repeated sections; no copy-and-paste code.
- Boy Scout principle: See that the code is in a better state than before this change.
- Check for debugging clauses and temporary remarks; reminders that programmers put into the source

10 Years Strong & Growing

Integrity, Efficiency, Service,
Guarantee, Global Productivity,
Midwest Rates, Universe,
UniData, D3, Barcode Services,
Nationwide Support, Web Services,
Custom Barcode Programming,
.Net Programming, Mentoring,
Project Outsourcing



www.pickprogram.com

contact@pickprogram.com

(614) 921-9840

Do you... have projects sitting on "the list" and not being completed?

Do you... just need some assistance from time to time?

Do you... want up-front, guaranteed estimates?

Senior Consultants are Currently Available.

"PICK Programmer's Shop has been outstanding to work with. They work with the highest integrity, are very responsive and quite knowledgeable. I can't imagine operating without them."

Patti Rowlette, Rowlette Executive Search

It's not just an anniversary. It's a guarantee.



Informer™
Quick and Easy Business Intelligence



The BI Survey 14 ranks Entrinsik Informer #1 in "Ease of Use" and "Chosen as Standard" categories among dashboarding & ad-hoc reporting vendors.

Find out why at www.entricsik.com

code, but then forgot to go back and address.

- Optimize if you can. If you know a more efficient way to accomplish something, share with the programmer.
- Check for standards. Does the new or changed section of this program match the previous architecture? Does it meet *generally accepted standards*? Does it meet *our company standards*?
 - Note that in order for the latter two to be considered you may want to spend some time documenting the company standards and make information about generally accepted practices available.
 - Some generally agreed upon standards both within and outside of Multivalue are
 - No (or limited) GOTO
 - Module re-use with GO-SUB
 - Minimize multiple and nested returns
 - Numeric vs word labels
- Establish naming conventions for variables, programs, and other parts of the software.

- Variables
 - The most important thing is to keep variable names unique. Don't use short variable names that may get reused — and their value reset — in other called sub-routines.
 - On the other hand, don't get carried away with a lengthy set of rules for naming variables — google *Hungarian Notation Abuse* for some amusing side effects of this effort. One example: Avoid prefixes that have contradictory meanings, are inconsistently used, and end up making the variable names unwieldy with no real benefit.
- Programs and other components benefit from the same guidelines, along with the additional consideration of how pronounceable the program name is. Imagine a conversation where you have to hiccup every time you want to refer to a particular routine.
- Casing: Uppercase, lowercase or camel case. We all have our preferences, but please let's agree on

something before we create a veritable quagmire of ASCII.

- Testability: Is the program ready to be tested?
 - Is there a framework for a test mode? For example, are there sections that display pertinent progress information if a debug flag is set?
 - Should your code review include reviewing against a test plan? Or reviewing the test plan itself?
 - Can any test code be replaced with the use of an existing API?

Once you've developed some standards it is possible to at least semi-automate some of the code reviewing functions. In PRC (gratuitous plug), a set of standards can be applied to code during the process to mark a project "development complete." Any failure to adhere to the automated standards can prevent the project from moving forward to the test phase, or whatever is next in the lifecycle.

This checklist is only a starting point. The key is to allow the checklists to evolve. Let people add, delete and refine. With their feedback also comes buy-in.

Armed with a great checklist, code reviews can be simple and quick while helping you drive up coding standards and avoid inconsistent quality. **IS**

SUSAN JOSLYN is the author and chief support person for PRC®, which is a Multi-Value based software development lifecycle management and IT Governance tool.

Thinking in APIs

BY KEVIN KING

Monolithic vs. API

There are countless programs and subroutines built for the telnet world. In some cases, these programs and subroutines can be thousands of lines long, many with multiple entry points and exit points. The great thing about this monolithic style is that once the program is running, it stays running until the program is finished. If the program needs to display a message, it can display a message whenever it needs. If the program needs to accept input, it can accept input at any time. When the program needs to calculate, it calculates. In the end, all the food groups of input, processing, and output are melded together into one big, monolithic meal.

Monolithic programming works great for telnet applications because once you're in the program, you stay in that program as long as it stays running and you have a solid telnet connection. For web-based applications however, there is no persistent connection and no way to keep one program running to do everything that needs to be done. For this type of programming, an API (application programming interface) may be a better approach.

The problem with historical code is that the lines have been blurred between what a subroutine does and what a program does.

With the API approach, subroutines are created for everything and each subroutine merely accepts input and produces output. API routines don't actually get the input from the user; they merely accept it. API routines don't actually display the output; they merely produce it for something else to display. A simple API routine might look something like figure 1. Notice that it does no INPUT of its own, no CRT nor PRINT; its job is simply to take the input that has been provided and produce the output. The caller of this routine can then capture input in whatever way it deems necessary and can display the output however it needs.

API routines are not specific to web programming, however – they can be called

both from telnet and non-telnet applications – as they have standardized inputs and outputs. Such a routine needs to know nothing about the environment in which it is called; it can take the input and produce the output completely unaware of its surroundings.

The problem with historical code is that the lines have been blurred between what a subroutine does and what a program does. Subroutines do more than simply convert input into output. Subroutines also prompt for input and produce output just like the programs that are calling them. In doing so, they are limiting their viability except in the interactive context in which they were first designed.

To get the most mileage out of subroutines, create API routines that avoid direct inputs and outputs. Subroutines should convert input to output and nothing more.

```
SUBROUTINE ADD.TWO (ANS, VALUE1, VALUE2)
*
ANS = VALUE1 + VALUE2
*
RETURN
```

Fig. 1

IT audits have you jumping through hoops?



PRC can help you meet your compliance requirements and make IT more agile and productive. No extra work, nothing to remember, nothing to fall through the cracks. Our software development lifecycle tool automatically prevents or detects change according to your criteria. You can deploy, rollback, test and report quickly, automatically and with confidence. Let PRC protect your company's valuable U2 data and software assets.



SJ+ Systems Associates • info@sjplus.com • http://sjplus.com



Lock with Care

One of the concerns with API validation like this is that at some point a record needs to be locked to be updated. Is locking a part of the validation, or is it a part of the update? The answer is, of course, “YES.” In other words, locking records for update is a function of both validation and updating.

In an API approach, the validation should be responsible for locking everything that needs to be updated. It should also be in charge of unlocking when everything *needing* to be locked can't actually *be* locked. This prevents traffic jams and deadly embraces. A record that cannot be locked is unable to be updated, and that should be as effective as any other error in stopping the update. This also implies that the validation API routine needs to communicate all those locked and loaded records to an update API routine somehow. The important part is that all of the validation is done first, and

Separate Validation from Updating

When writing telnet-based screens, it's not uncommon for validation and updating to be intermingled into one coagulated mess. Start the validation, and if everything looks good, do some updates. Check a few more things, do a few more updates, rinse, repeat.

Of course, nobody we know would do this, right? Not only does this contribute to inconsistent updates, but it makes debugging harder (i.e. “How did the invoice get created without the G/L being updated?”). It prevents using the same routine in both a stateful and stateless environment.

Instead, we can write updates as API routines. Because these routines will focus on a single task — validation *OR* updates *OR* transformations — they are less likely to fail. However, if they do fail, they can fail gracefully, rolling back any changes that may have been made. Our separate API routines for validating will return information back to the caller about what is needed for the updates to be successful. The point is that by separating the parts, you can call a validation routine as many times as needed (and from any interface) and get a proper answer each time. Yet the saved information has not been af-

fectured because the updates are in a different routine. Only when everything is assured to update without incident should the update routine be called. For example, a simple validation API subroutine might look like Figure 2.

Also, consider that “update” in this context may not actually involve a change to the database. It may simply involve transformation; changing a value in a memory variable. Makes no difference, all validation should be done before any updates occur.

```
SUBROUTINE VALIDATE.UPDATE (STATUS, KEY, RECORD)
*
... extract variables from the record and
... read anything else that might be needed.
*
STATUS = 'ERROR'
IF (INV.DATE EQ '') THEN
  IF (CUST.IS.ACTIVE) THEN
    IF (CREDIT.LIMIT.IS.OKAY) THEN
      STATUS = 'OK'
    END ELSE
      STATUS = 'NOLIMIT'
    END
  END ELSE
    STATUS = 'INACTIVE'
  END
END ELSE
  STATUS = 'INVOICED'
END
*
```

Fig. 2

once the update is assured, the update can proceed.

Managing State

In a telnet environment, we don't even think about managing state as we can hold all our variables in memory for as long as we need (or until a road crew "inadvertently" cuts our network connection). In a web environment, we can worry less about the road crew as there's no persistent connection to be lost. However, managing state is much more important in a web context to keep a conversation going between the user experience and the back-end processing.

So who is responsible for managing state? The short answer is that it shouldn't matter. When constructing applications using the API method, the same routines can be called from any number of interfaces, both state-

ful and stateless. As long as each routine receives the input it needs, it can produce the output that the caller requires. With these details out of the way, managing state becomes merely a detail in the process of constructing interfaces to the API routines.

Leverage the Adapter Design Pattern

Once the API routines are all in place, you can "adapt" the inputs and outputs from various contexts as needed. One adapter (calling routine) might be for the telnet application whereas another supports web, and yet another might be for a service architecture using a drop-file interface. The point is that the interface itself doesn't matter until you create the adapter.

You can create as many different interfaces into the application as needed, all using the compendium of API routines previously implemented. In the end,

the adapter will hold the unique part of each user experience. All that logic in the API routines will be leveraged by each interface by flowing through the adapter. The rest of the application — daresay the *majority* of the code as implemented into API subroutines — can be reused by all of the interfaces.

There are several other issues to keep in mind when creating a "write once, use everywhere" application, but hopefully this establishes a foundation on which to begin the discussion. **IS**



KEVIN KING is the President and Chief Technologist with Precision Solutions, Inc., a leading technology solutions pro-

vider in Longmont, Colorado. He can be reached by email at Kevin@PrecisOnline.com or by voice at 303/651-7050.



taking multivalued ...
where it has never been before

Ladybridge Systems Ltd

17b Coldstream Lane, Hardingstone, Northampton, NN4 6DB, England

Worldwide distributor: Zumasys, 9245 Reasearch Drive, Irvine CA 92618, USA

www.zumasys.com

- Close compatibility with most other multivalued environments
- Easy migration process
- Maintenance-free file system for ease of use
- High quality documentation
- QMClient API for development of GUI and web applications
- Low licensing cost
- AccuTerm bundled at no additional cost
- Many unique features

www.openqm.com

FROM THE PRESS ROOM



Kore Technologies Announces Release 5.4 of Kommerce Server eCommerce Suite

Kore Technologies announced today that Release 5.4 of its integrated platform, KommerceServer eCommerce Suite is now available. The new release is up to 20 times faster when searching for products. KommerceServer's new B2B and B2C features are designed to enhance the user experience and improve product information management.

With the new features of KommerceServer Release 5.4, users can:

- Use the improved quick search to find more matching products sorted by relevance
- Search for available inventory by branch location
- Search for nearby store locations based on zip code

The new release enables eCommerce administrators to:

- Promote the website through targeted coupon marketing
- Define zones and rates to better manage internal truck fleet deliveries
- Implement customer loyalty programs by generating leads for key affiliates
- Use a request for quote strategy to engage the salesforce and not disclose pricing

"Nearly all of our enhancements are customer-driven so the design is vetted to address real-world business scenarios," explains Ken Dickinson, co-founder and managing partner. "Manufacturers and distributors need to manage product information and produce engaging content to drive traffic to the website and physical store."

KommerceServer eCommerce Suite is an award-winning B2B and B2C eCommerce platform which includes a storefront, customer portal and mobile edition, and is engineered to integrate with any back office ERP system (e.g., Epicor Eclipse, Prelude, Infor LN and more). KommerceServer's product information management and content management systems integrate product data from multiple sources to simplify website creation and administration.

About Kore Technologies

Kore Technologies is a leading provider of enterprise

integration, business intelligence and eCommerce Web solutions for MultiValue and Microsoft SQL Server databases. Founded in 1999 as a professional services and software development company, Kore specializes in providing business-to-business connectivity and best-in-class product integration with its flagship product — Kourier Integrator. Leveraging this technology, and Kore's extensive experience with enterprise applications and database systems, Kore delivers solutions for integrating disparate applications, building data warehouses, and developing business intelligence solutions.

For more information, visit <http://www.koretech.com> ■



Universal Driver 5.0 to be Made Available for use With OI 9.4

Revelation Software is pleased to announce the features and benefits of the Universal Driver 5.0 that will be available for use with OpenInsight 9.4 in October 2015!

The UD 5.0 will be included free of charge with the pur-

chase of OpenInsight version 10 licenses.

- Latest communication technology : The Universal Driver 5.0 utilizes the latest communication libraries to provide robust, reliable service between the UD server and its clients.
- Secure communications option : If desired, the communications channel between the UD server and its clients can be SSL encrypted, preventing unauthorized interception of sensitive information.
- Authenticated clients option : If desired, only explicitly authorized client software can connect to the UD server over a secure SSL channel.
- Connection retry : If the communication link between the client and server is temporarily lost, and then restored, the client will attempt to reconnect and resume normal operations.
- VSS friendly : The UD 5.0 is VSS (Volume Snapshot Service) aware and can thus be backed up with minimal disruption to normal operations.

About Revelation Software

Founded in 1982, Revelation Software delivers a suite of application development tools and companion services that take full advantage of leading network computing architectures, messag-

FROM THE PRESS ROOM

ing, groupware, and client server platforms. Today, the company's flagship product OpenInsight is the only database development and application environment that provides both Windows and Java-based GUI tools to develop and deploy web-based and client server applications that support native and relational XML, SQL, Lotus Notes and the leading legacy Multi-Value data sources such as Arev, Pick and IBM Universe. There are more than 1.5 million licensed users of Revelation products across 60,000 deployed sites worldwide. The company has offices in Westwood, New Jersey, as well as a European distributor in the United Kingdom, and an Asia Pacific subsidiary in Australia.

For more information, visit <http://www.revelation.co> ■



XLr8Editor for D3

U2logic is pleased to announce that XLr8Editor is out of beta and ready for general use for D3 database users. XLr8Editor is based on Eclipse IDE, which is used by millions of programmers. This interface requires the installation of MVSP from Rocket

Software.

XLr8Editor for D3 features:

- Continuous compile which compiles your code on the server and returns the resulting errors as you type
- Eclipse search engine that allows regular expressions
- Off-line mode to make changes without a database
- Eclipse Task Tag system to bookmark your code with TODOs or FIXMEs that reside in your workspace
- Eclipse Version control for programs, dictionaries, and Procs using GIT, SVN, and Team Foundation
- XLr8 Synchronization engine that allows you to sync your local copies with the database
- Colors for keywords, Labels, Comments, Strings, Numbers, and Punctuation
- Automatic comparison between your local copy and the database copy
- Automatic save at specified intervals

About U2logic

U2logic has been moving forward ever since the Web came to being. From our early attempts at CGI code, to using RedBack IDE, and now to our current AJAX open source approach, U2logic has always been on the forefront of web-based technology. This technology is created for Universe and Uni-data Rocket Software's U2 databases.

U2logic has been gearing up two main goals:

The first goal is to bring to the IBM U2 databases many of the same tools the rest of the programming world enjoys. U2logic has based its entire tool set on Eclipse. The many products U2logic has developed fill those ancient holes that were part of U2logic beginnings in the early 1980s. U2logic's family of tools is called XLr8.

The second is to create applications that allow users to experience the U2 databases without realizing what database they are using. Beauty is in the eye of the beholder, that is why our applications are very easy on the eyes. All of our applications, whether data entry or reporting, are seamlessly tied to open technologies such as Apache Tomcat, Firefox, Job Scheduling, and AJAX without the user ever knowing it.

For more information, visit <http://www.u2logic.com> ■



Entrinsik Announces Listing on GSA Federal Suppliers List

Entrinsik Inc. announced that it is now listed on the U.S.

General Services Administration (GSA) IT Schedule 70. The IT Schedule 70 is a list of vendors the GSA has pre-authorized to provide federal agencies with IT equipment, software and services.

The GSA is America's only source *solely* dedicated to procuring goods and services for government agencies. Vendors must pass through an arduous and lengthy review process in order to be approved for GSA listing, enabling buyers to save time and reach better purchasing decisions by choosing from the pre-approved list of suppliers.

The GSA schedule contract allows federal, state and local government agencies to purchase Entrinsik Informer software and services using terms and prices that were previously negotiated. Entrinsik Informer is available on the GSA IT Schedule 70 under contract # GS-35F-442CA.

Entrinsik Informer is highly ranked by end users as being an intuitive, easy-to-use reporting and business intelligence solution providing access to real-time data from disparate sources, multi-dimensional data analysis and drag-and-drop dashboarding capabilities. Informer's award-winning features are securely accessible anytime, anywhere from an end-user friendly web-based interface.

FROM THE PRESS ROOM

Entrinsik customers include the US Navy, over 350 universities and colleges, USAA, the State of Connecticut, and Emergency Response Management in Mexico and USA.

"Entrinsik is leading the way in terms of setting the standards as to what dashboard and ad-hoc reporting environments need to adopt," says Doug Leupen, CEO and President of Entrinsik. "We are excited about our new listing on the GSA schedule and we look forward to delivering significant value to agencies who need an agile, afford-

able, easy-to-use reporting and business intelligence solution like Informer."

Agencies acquiring Entrinsik software through the GSA schedule contract will benefit from a significantly reduced acquisition timeline, pre-negotiated rates and terms, and can circumvent the sometimes cumbersome RFP/approval processes, all of which make selecting Entrinsik appealing to government decision-makers.

About Entrinsik

Entrinsik, Inc. is the market leader in providing web-

based operational and analytical reporting tools to the MultiValue marketplace. Today, with Entrinsik Informer, thousands of end-users using MultiValue databases enjoy the ability to leverage the Web to share information assets with virtually unlimited number of people anywhere at any time. With its intuitive, easy-to-use interface, Informer provides end-users, IT staff, and executives a business intelligence environment and consistent, real-time information impacting key business processes. Spectrum Maga-

zine has described Informer as a "supercharged reporting tool...designed for decision-makers."

Founded in 1984 and in the MultiValue market since, Entrinsik aims to provide leading edge technology and unsurpassed service to their customers.

For more information, visit <http://www.entrinsik.com> ■

eXtra Newsletter

Stay on top of Industry News

- ◆ Tech Tips
- ◆ Job Postings
- ◆ New Products
- ◆ Corporate Updates



www.intl-spectrum.com/newsletter

OpenInsight

Data Encryption at REST

OpenInsight incorporates a new encryption service that system administrators can deploy to encrypt (on a field by field level) data in OpenInsight. The data will be *encrypted at REST* (Representational State Transfer) using industry standard encryption routines. Clients and the encryption service use the Windows Communication Foundation (WCF) to communicate, thus allowing the data to be encrypted in transit. Through the use of an Encryption Server (normally, but not necessarily, the same machine as the Universal Driver server), each OpenInsight workstation seamlessly and securely encrypts data for storage when written into an encrypted volume.

The metadata and actual table contents are stored on the encryption server using an AES encryption algorithm with a key size of 256 and two key iterations. For more information on this type of encryption see:

http://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Configuration of the Encryption Server

One or more Windows Groups are assigned to one or more Encryption Server Roles provided with the Encryption Server. The people who can

The PASS volume can be moved among different Encryption Server systems, and is thus suitable for transporting data either temporarily or permanently.

assign groups to the other roles must be members of the Role Manager groups. Members of the groups in the Server Manager roles can change the encryption settings on the server. Table Manager groups contain the people who can define or change the encryption settings on tables. Members of the groups that are in the Assign Impersonators role can programmatically set which group they wish to impersonate.

Impersonators are used when multiple different permissions are required, but all users log into Windows as the same user (e.g. the OpenInsight for Web user). Members of the groups that are in the Data Access roles are allowed to access the encryption system. Any users who are not members of these groups will not be able to connect to the encryption server.

Different Types of Encrypted Volumes: Site-specific and PASS

There are two different types of encrypted volumes. A normal Revelation

Data Encryption at REST (RTIDER) is a site-specific volume that uses encryption keys which are maintained by the Encryption Server. All workstations accessing this Encryption Server can, with proper permissions, access the data. However, the tables contained in these volumes cannot be used by any other system running its own Encryption Server.

Alternatively, a volume can be defined as an RTIDER PASS (Password Authenticated Single Session) volume. To access tables in a PASS volume, the user must enter the required passphrases during table attachment. The PASS volume can be moved among different Encryption Server systems, and is thus suitable for transporting data either temporarily (for example, when copying tables between fixed systems) or permanently (for example, putting such tables on a thumb drive).

Creating a Site-specific Volume

Before encrypted tables can be created, an encrypted volume must first be defined. For a normal RTIDER site-specific volume, the CREATE_VOLUME procedure is used, with additional encryption parameters passed in as part of the "control_location" parameter:

```
Create_Volume( volume_name ,
```

```
filing_system, location, control_location, status)
```

The volume_name is the name of the new volume to create; the filing_system must be "RTP57E" for an encrypted volume; the location is the physical Windows directory for this volume; and the control_location contains the encryption details, in a @VM delimited string:

```
server_address : @VM : server_port : @VM : show_protected
```

The server_address is the IP address of the Encryption Server. The server_port is the port number that the Encryption Server service is "listening" on. A show_protected value of 0 indicates that encrypted fields should be blanked-out for users who do not have permissions to decrypt those fields. A value of 1 will display them in their encrypted format to those same users (Fig. 1).

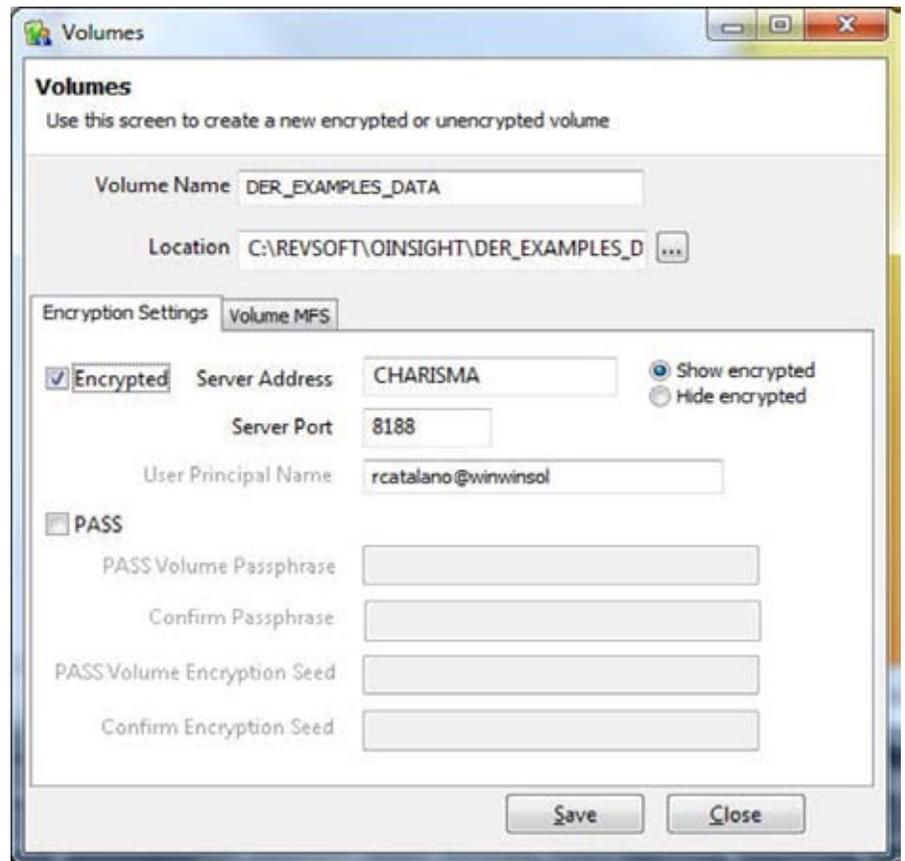


Fig. 1

When was the last time you read something fun?

Bottom line: Tiago and the Masterless is a book that only a programmer could have written, but not one that only a programmer can love.
- Jon Frater's review on Amazon.com

Upcoming
Tiago in the Tunnels of Krall
Tiago faces the Abyss

<http://www.hdwpbooks.com/books/tiago>

Creating a PASS Volume

To create a PASS volume, you must use the CREATE_VOLUME_PASS procedure:

```
Create_Volume_Pass( volume_name,  
location, control_location,  
passphrase, initVector, status)
```

The volume_name, location, and control_location are defined as above. The passphrase and initVector are the unique pieces that will be used to encrypt the information in this volume. The passphrase and the initVector are both user-defined strings or words; they should be unique to each PASS volume, but may be anything that is both memorable and easy to keep secret.

Note that, since PASS volumes require the passphrase and initVector to be specified *each time* they are initially accessed, you should NOT add tables

(files) in the PASS volume(s) to your database definition. Instead, you must always run ATTACH_TABLE_PASS to attach, *on a per-session basis*, the tables in the PASS volume (Fig. 2):

```
Attach_Table_PASS( volume_  
pointer_name, tablelist,  
database, passphrase, initVector,  
status)
```

Creating Encrypted Tables (Files)

Tables in the encrypted volume can be created using the CREATE_TABLE procedure, which has now been extended to include an additional parameter:

```
Create_Table(locationlist,  
tablename, dictflag, databaseID,  
tableattributeslist, protectflag,  
unused, unused, encryptInfo)
```

The encryptInfo parameter defines which fields in the table should be encrypted, which groups should be able

to access the data in the table, and how keys should be processed. This parameter is an @FM delimited string containing, in order:

1. list_of_fields
2. list_of_groups
3. error flag
4. key part flag

Field 1 (list_of_fields) contains a list of which fields should be encrypted. This list is comma-delimited, and may indicate the keyfield (0), a part of the key (0*<keypart> for example: 0*1), individual fields (by number), or indicate all fields, including the key field (-1). All (-1) is obviously exclusive. You can list individual fields "0,1,3,4", or "-1", but you cannot mix the two.

Field 2 (list_of_groups) defines which Windows groups should have access to

Feedback

*What came first,
the letters or the
letters-to-the-editor
department?*

International Spectrum Magazine has a Feedback Department, sometimes known as Letters to the Editor.

We want to hear your comments, your reactions, your agreement or disagreement with what you see. Also, do not hesitate to let us know about things happening in the MultiValue Community we may not have heard about yet.

Please send your comments by e-mail to:
editor@intl-spectrum.com

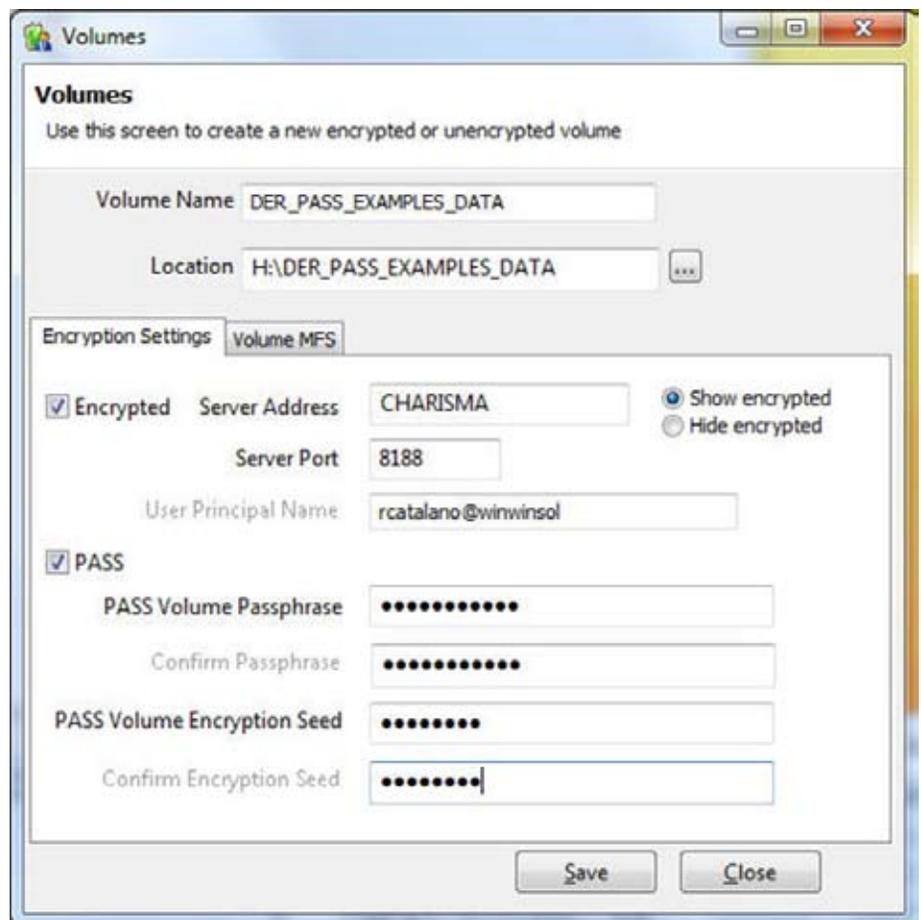


Fig. 2

each of the encrypted fields. This list is in the format <fieldnumber>: <groupname> {; <groupname>; <groupname>} {, <fieldnumber>...}. For each field that may be encrypted (as defined in the list_of_fields), you may specify one or more Windows groups that can access that fields' information. As above, the field number may contain "0", "0*<keypart>", field numbers, or "-1". The group names can also be "*" to indicate "all valid Windows users in this domain." An example of the list_of_groups may be "0:inventory;acctg,1:*,3:acctg,4:inventory". To cover everything for everyone use "-1:*".

Field 3: If a user attempts to update a record for which he/she does not have full permissions, the Encryption Server can either "merge" the original record's fields with the updated data, or generate an error. If field 3 (error flag) is set to "1", then an error is returned; if set to "0", then the modified data is merged with the protected fields.

If a key contains multiple parts (separated by the key delimiter "*"), it can be encrypted as a single piece of information, or each key part can be encrypted separately. If field 4 (key part flag) is set to "1", then each key part is encrypted separately; if set to "0", the entire key is treated as a single string, and the key delimiters are ignored during the encryption process.

Please note that it is *specifically allowed* to create a table with no encryption information in an encrypted volume. Simply pass an empty string ("") as the encryptInfo parameter. If instead you wish to create an encrypted table with the default settings, pass in a parameter with 4 fields, each of which are the empty string ("") (Fig. 3, Fig. 4, Fig.5). **IS**

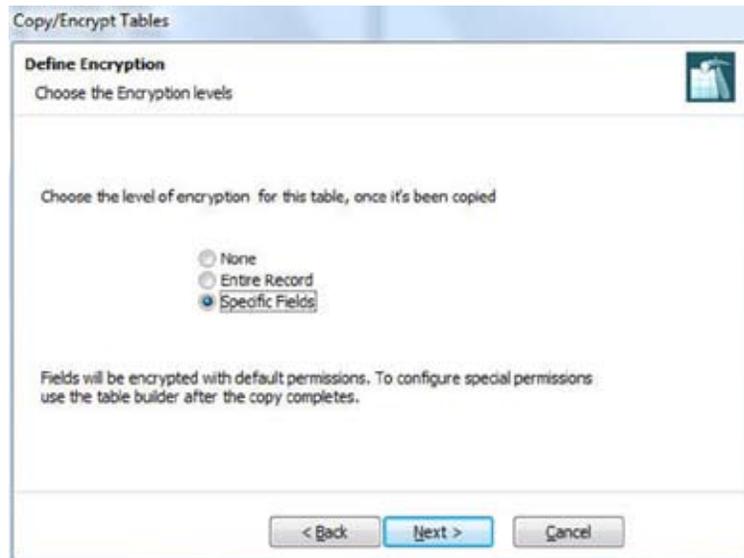


Fig. 3 Choose your level of encryption

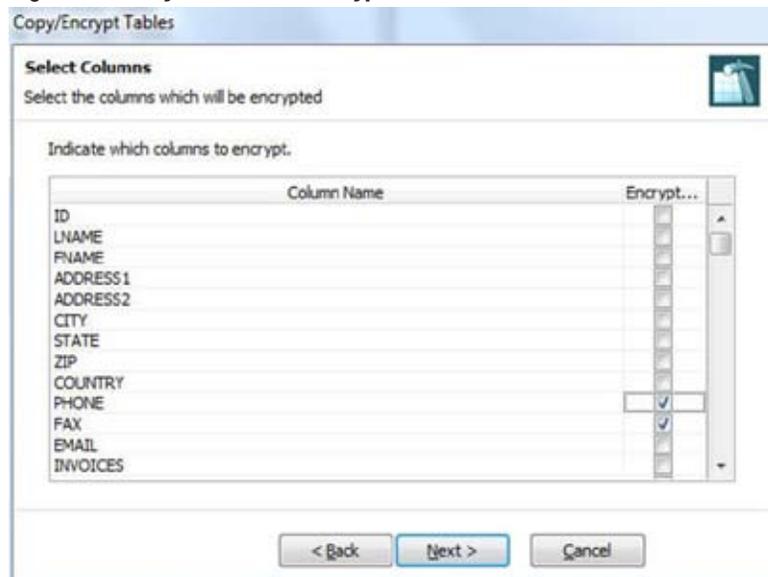


Fig. 4 Select the columns (fields) which will be encrypted

ZIP	COUNTRY	PHONE	FAX	EMAIL	INVOICES
1	97675	(201) 722-9814		mike@winwinol.com	
2	54321-2345 United States	(301) 555-1212		jose.canusee@barner.com	
3					
4	44334 US	(800) 555-5555			
5	44334 USA	555-5555			
6	98543	(716) 323-3232			
7	98965 USA	(201) 555-1234		ben@bouncers.com	
8	10821	(212) 999-8888			
9	32121	555-5555		dee@happiest.com	
10	55001 USA	(321) 444-3212			
11	93212				
12	07675 USA	(201) 666-9682		lee@theirhorse.com	
13	92870 US				
14	43556 USA				
15	04086 US				
16	94407				
17	08018	(973) 562-4234			
18	10675 United States	(202) 664-1234		hjohnson@GCE.com	
19	10023 United States	(212) 798-2100			
20	06902 United States	(203) 359-9988			
21	08217 United States	(856) 959-7757			
22	07662 United States	(301) 843-7600			
23	07948	(511) 828-4588			
24	92618 United States	(949) 442-4400			
25	98104 United States	(206) 789-8313			
26	99362 United States	(309) 525-2166			
27	80302 United States	(303) 513-8877			
28	77077 United States	(281) 752-4748			
29	07675 US	(201) 664-8764		btruman@truman.com	
30	95967 United States	(818) 674-3654			
31	07760 United States	(732) 842-6145			
32	92691 United States	(949) 380-1668		info@nebula-rmd.com	
33	10033 United States	(201) 444-8764		truman@truman.com	

Fig. 5 Users who are not members of this group cannot view the FAX data

Why I Hate Technology

BY CHARLES BAROUCH

I hate technology. You are the face of technology. Therefore: I direct rage and contempt upon you. If you aren't the hands-on support person, ask your hands-on support people about this. Part of the disconnect between IT and the rest of the company comes from the sense that we are the personification of tech.

When I'm in the hot seat, doing support, I get questions like "Why are we using Outlook? My friend's dentist's company uses Google Mail and they don't have these problems." One of the things we need to read into this question is the assumption that we picked the software in use. A lot of people in your company will blame you for the fact that the company has whatever program or process it is that you are fixing at the moment. My favorite is when someone with twenty years in does that to me when I've been on the job for a week. They know intellectually that I couldn't have been a decider, but emotionally, for them, I am the technology.

Likewise, when I help someone who just got a bigger monitor or a faster system, I'm their hero; even if I'm just there to load AccuTerm. For most users, IT is an amorphous blob. Being part of it implies that we are experts in every bit of it and that we are the ar-

For most users, IT is an amorphous blob. Being part of it implies that we are experts in every bit of it and that we are the arbiters of which hardware and software have been selected.

biters of which hardware and software have been selected.

While I can't give you everyone's list of pet peeves and genuine hot button issues, here's a non-definitive sampling from my list of techno-hate. And, for the record, *I don't blame any of you for these.*

10. Tech is Destroying English

I verb my nouns now. I noun my verbs. Backup used to be two words, and it was used as a verb, not a the name of a process. I use abbreviations like VPN (Virtual Private Network) and SSH (Secure SHell) as if they were verbs.

We end things by hitting Start. We execute things. And anything we don't screw up, auto-correct screws up for us.

On top of this, we've gotten the whole world to speak differently. A friend, Charlie Hoover, recently asked a group of us online to come up with a

sentence that makes sense today, but which would have made no sense to us twenty years ago. Here's one: *Why does the calculator I'm downloading to my phone need in-app purchase permissions?*

9. Three Screening

When I was a kid, I'd talk during the TV shows and sit in rapt attention for the commercials. Now, between my laptop, TV, and phone, too much of my life is spent staring at a screen. We've become a culture of zombies thanks to our always-on tools and toys.

To be fair, I'm a willing zombie. I just put down my smartphone after returning a call, playing a game, and checking my calendar. Now I'm on my laptop, writing this, before VPNing into a client to fix a data transfer problem. After that, I might catch up on a DVR-ed TV show or two while I finish that next step on the app I'm developing.

8. No One Reads (Because They are too Busy Reading)

Reading for pleasure seems to be a lost art. Why don't people have more time for literature? They are too busy reading (Facebook, G+, Twitter, LinkedIn, et al.) to have time to read. When the social media isn't demanding our attention, the mountains of data at work does. The fact that I can now read a

virtual version of a thousand-page report on my phone ends up obligating me to read a thousand-page report on my phone.

The boundaries between work time and private time are eroding. Leaving the office used to mean leaving access to mountains of paper. You couldn't be expected to bring all of that home, to sift through that much data. Now, the tools we have made it possible. Impossible had a better quality of life.

7. Glut

I speak fluent Proc, mvBASIC, Delphi, PHP, Javascript, etc. I'm passable in the C-Family (Java, C, C++, C#, Objective-C). I'm learning Python. I keep wanting to learn Dart but there aren't enough hours in a day. One size does not fit all; but seriously, how many ways do we need to write *Hello World?*

There's also a format glut. Recently, I had to take Excel's (proprietary) format, convert it to (a semi-proprietary implementation of) XML, merge the results with a CSV, and validate it against a MultiValue PRODUCT table. If they'd thought to add one more format, perhaps a MySQL database to the mix, you would have heard my head explode. And this wasn't the only data transformation project I was working on that week.

Part of my livelihood comes from this chaos. Despite that, even I am tired of it.

6. Pronouncements of Death

MultiValue is dead: false. Mainframes are dead: false. Cobol is dead: false. Can we please stop bashing every tech that we feel might challenge the tech we like? As a technology journalist, I am tired of a whole category of technology reporting. Please stop telling

me what six minutes worth of research — or less — has led you to conclude.

There are still millions of lines of counted flat files in use every single day. Very little in tech ever completely dies off. Photography didn't cause us to stop making oil paints and canvases. TV didn't close down every movie theater. New tech may cut into the popularity of — or even marginalize — older tech, but that's not the same as dead.

5. Techno Consumerism

Now that I no longer carry an iPhone, I find myself looking for an excuse to buy an iPad. My Android tablet is getting slow, so I want a new one. I own two (current) laptops but I still find myself looking at the sales. Now, if you'll excuse me, I still haven't finished tweaking my Raspberry Pi.

4. Electric Bills

I routinely run two laptops, a phone, a Bluetooth earpiece, a tablet, and a TV. They require a DSL modem and router. And I'm not the only person in my house. My carbon footprint is embarrassingly huge.

3. Politics

This isn't strictly a tech complaint. However, technology has made it possible for us to have a twenty-four-hour news cycle and the twenty-four-hour rumor mill. I can now read something that ticks me off about politics *any time of the day or night*. Even better, I'm no longer restricted to being upset about the U.S. elections because I have access to articles about political situations in every country in the world.

I'm not sure why I need to have an informed opinion on Toronto's ex-Mayor, or Australia's PM, or any number of people in the governments of England,

Germany, and Greece, but I certainly have access to enough information if I wanted to form those opinions.

2. The End of Privacy

I look at a circular saw online. Next six webpages I go to, all of them about writing, suddenly have ads about — you guessed it — power tools. There are algorithms out there that think they know us. And, while they are sometimes right, well, let's just say that Amazon suggests books to me which I've written. Granted, I am interested in the books I write, but, by the same token, I probably have those books already.

Facebook stalks me. Google still wants to sell me that thing I looked at six months ago. Apple is sure that they have my musical tasted perfectly profiled. Netflix just suggested something to me that I'd never watch in a million years. We used to see the Internet as an anonymous land. All illusions of privacy are gone.

When you see an article about the NSA collecting phone records, remember that in the early days of telephones it would have been impossible. Technology is why they can sweep all that data together. The candle shop in 1885 couldn't analyze my buying trends. Now they can pay pennies per thousand clicks to suggest something I'm statistically expected to want.

And, for the record, "If you have nothing to hide, you have nothing to fear" is literally lifted from the Nazi propaganda machine. The re-emergence of comments like that terrifies me.

1. Windows 10

My new number-one hate is the way Microsoft is handling Windows 10. Mandatory updates concern me. Presently, if an update breaks an applica-

tion, the vendor can warn you not to take it. With the new Windows, you'll be forced to take an update even if it puts you out of business.

I've lost work this week to Windows deciding that their update was more important than anything I was doing. I'd hit return in a word processing document, only to find that a 'reboot now' window had popped up and stolen that return. Nothing like a shutdown without an opportunity to save my work.

In my mind, if I paid for this laptop, I get to decide what the priorities are. A world where the O/S is in charge doesn't work for me.

This is why I've stopped using all technology. Except for my phone. And Syd and I need the TV (and DVD player) to catch up on Game of Thrones (no

spoilers, please). And I just leveled up another character on Marvel Heroes, so I guess I need the laptop and the Internet. Maybe I should tell everyone on Twitter about how I've given up tech. I'll get to that after I install additional software that I need to support my new client. ::sigh:: **IS**



CHARLES BAROUCH is the CTO of HDWP, Inc. and the Publisher at HDWPbooks. You can read his writing in

International Spectrum, Theme-Thology, Novo Pulp, PerhelionSF, and the Interrogative series, which begins with Tiago and the Masterless.

can
MultiValue can't do that.

Change your thinking.

Modern MultiValue, LLC
Modern Solutions for Modern Applications
info@ModernMultiValue.com
www.ModernMultiValue.com
+1.225.341.1778

WANT TO SEE A SPECIFIC TOPIC?

International Spectrum is looking for writers, feedback, and topic ideas. We all have specific topics and issues that we need answers to find solutions for. Send us an E-mail with topics you would like to have covered in the magazine or on the website.

E-mail: nathan@intl-spectrum.com

WANT TO WRITE?

Expand your professional credentials, and provide us with an article.

Give us a rough and ugly outline, and we will help you refine it, proof it, and make it press ready. Or you can give us something polished, proofed, and press ready to publish.

Share your thoughts and expertise with over 10,000 fellow MultiValue developers and users.

E-mail: editor@intl-spectrum.com

NEED A MENTOR?

Mentors give developers the ability to ask industry experts for direction, code examples, and/or just ask them to see if something makes sense. Sometimes, all you need is a resource or example to start or complete a project.

Check with us to see who is available for mentoring, and how you can take advantage of it to save your business or company money.

E-mail: nathan@intl-spectrum.com

WANT TO BE A MENTOR?

We have many retired or semi-retired professionals out there that would love to share their knowledge of MultiValue development. If you are one of them, please contact us to see what mentoring is all about.

E-mail: nathan@intl-spectrum.com

MARKETPLACE

ACCOUNTING

Natec Systems

www.natecsystems.com | nrector@natecsystems.com

mvQB QuickBooks API for the MultiValue Database

- Read/Write Directly to Quickbooks Databases
- mvQB API is Designed for the MultiValue Program to Use
- No Need to Learn the Internals of QuickBooks
- QuickBooks Pro/Premier/Enterprise

Phone: 303.465.9616
E-mail: mvqb@natecsystems.com
Website: www.natecsystems.com

NATEC Systems
Providing Solutions to your MultiValue Questions

COMPLIANCE

SJ+ Systems Associates

www.sjplus.com | sjoslyn@sjplus.com

CONSULTING

Drexel Management Service

www.drexelmgmt.com | dconboy@drexelmgmt.com

Execu-Sys, LTD

www.eslly.com | mh@eslly.com

HDWP

www.HDWP.com | results@HDWP.com

J Squared Software, LLC

www.jsquaredsoftware.com

30+ years experience Business Applications & Support

MultiValue platforms - per diem or fixed price

Juddlove@jsquaredsoftware.com | 203-325-0275

Modern MultiValue, LLC

www.ModernMultiValue.com | info@ModernMultiValue.com

PICK Programmers Shop

www.pickprogram.com | brian@pickprogram.com

Precision Solutions

www.precisiononline.com | Kevin@PrecisOnline.com

DATABASE

Ladybridge Systems Ltd

www.ladybridge.com | sales@Ladybridge.com

REPORTING

Brian Leach Consulting, LTD

www.brianleach.co.uk | brian@brianleach.co.uk

Entrinsik

www.entrinsik.com | sales@entrinsik.com

TERMINAL EMULATOR

Zumasys

http://www.zumasys.com/products/accuterm/

AccuTerm
by Zumasys

AccuTerm® software, the leader in terminal emulation, allows you to access your MultiValue application—whether it is on-premises or in the cloud—from any Windows device.

www.zumasys.com/accuterm
818-951-1891

HIGH IMPACT 4-COLOR DISPLAY AD

Data Systems, Inc.
Sophisticated order processing and fulfillment systems

303.333.3333 info@example.com www.example.com

Cost Per Year

1"	\$1,811.00
2"	\$3,623.00

BASIC LISTING: \$690.00

DataSystems, Inc. http://www.example.com, ... 333.333.3333, ext. 1

- Includes company name, website, and phone number
- Approximately 60 characters per line
- Place under category of your choosing

ADDITIONAL LINES: \$280.00 per line

DataSystems, Inc. http://www.example.com, 333.333.3333, ext. 1
Sophisticated order processing and fulfillment systems

- Approximately 60 characters per line



New Blood Part 4

BY CLIFTON OLIVER

Recap

In the previous three parts of this series we've talked about the need to attract developers from the "Millennials" in order for the MultiValue database to survive. Part one discussed the problems we, as the MultiValue Community, create for ourselves when we don't think before we post in public forums: The damage done by MultiValue developers when acrimoniously airing our own opinions — supported by facts or not — about what *isn't right* about this MV thing. In part two, we went on to explore a bit about what it takes for an IT shop to attract Millennial-aged developers and what it takes to keep them.

Part three was probably the most likely to cause roars of disapproval and outrage. Frankly, it is a topic that I've been on about for years now and still causes traditional MultiValue developers to grab torches, tar buckets, and chicken feathers and head in my direction. I'm talking about the extremely simple expedient of *not coding in all uppercase* like the 1960s and '70s. The style has changed and we need to change with it.

"The Millennials are Coming!" is no longer true. The Millennials are already here, folks.

Onward

Well, this final part in the series shouldn't be as "personal" a challenge since many of the things we will touch on are not currently being done in most MultiValue shops. That's not to say that they will be accepted. After all, the attitude of "we don't do things like that because MultiValue doesn't have to" is the usual comeback from the Cowboy Coders. But they aren't the ones who are making the company decisions and writing the checks that pay our salaries, consulting invoices, license fees for vertical applications, etc. If you weren't at the opening of this year's (2015) International Spectrum conference, you might want to sit down. This might come as a bit of a shock to you other old-timers (I include myself, and it was).

Nathan pointed out that depending on what birth date range you use to define that vague "Millennials" term, "The

Millennials are Coming!" is no longer true.

The Millennials are *already here*, folks. They are quickly moving up into IT management. They'll be making the decisions about which databases, which tools, which developers the company is going to spend money on. And what gets thrown on the garbage heap.

Question: What's the "Right Way" to develop software?

Answer: The way the 32-year-old dude(tte) who is *one, two, or more levels above you* in the org chart says.

What does their idea of a modern development shop look like? It varies. But in most cases it contains a mix of the following trends.

Use of Modern Tools

Got news for some of you: The ED line editor is not a modern tool. If you aren't at least using a GUI editor (read *real*, not an old green-screen thing in a telnet session) you have a bulls-eye painted on you. I am absolutely stunned when I keep running across MultiValue developers still writing code with the ED editor and its variations. There are a number of great pro-

gram editors out there all the way from free to “not much” in cost.

IDEs are even better. No modern developer codes, then runs, then debugs at the command line. (Though Python might be an exception. I’m still looking.) This is an area where, in my opinion, MultiValue still falls extremely short. It’s getting better with some of the platform enhancements and third-party products, though. More on that in a technical article, I think.

Still doing all your reports with ENGLISH/RetrieVe/INFORM etc.? It still has its place. But Millennials expect modern reporting options. If all they see is a carry-over from the green-bar paper days, they’re not going to be interested in working with it or keeping it around. Time to branch out and explore some of the other options.

The same thing holds for integrating MultiValue systems into the larger IT ecosystem of the enterprise. Too often traditional MultiValue developers haven’t moved past the idea of bulk import and export of comma (or pipe or tab) separated flat files. And when the “mainstream” side of the house wants to get or send data or processing requests as XML, they are dismissed with a curt, “MultiValue doesn’t do XML.” (Or web services. Or JSON. Or...)

So. Tools. Get away from the TCL prompt and learn what’s out there. That doesn’t mean you need to learn how to call a SAX parser to process an XML document that won’t fit into memory under the DOM model common in the MultiValue Basic extensions of some of the platforms. But please learn what sentences like that *mean*. And learn when those tools

should be used. And when to call on someone else. There is no shame in knowing when to punt. Your Millennial VP will make note of that and thank you for it.

Use of Modern Methodologies

There are few MultiValue shops I’ve visited that have a formal testing methodology established. And by testing methodology, I am not talking about the simple, “check it out, test it, check it in” source code control — though even that is somewhat rare (but getting better, I might add). I’m talking about a catalog of established, predictable tests against a known test database with *predictable* results. When you don’t have that, you aren’t *testing* your changes; you’re *trying* your changes.

So what is it that the Millennial VP expects to see (at a minimum) in a modern development shop? Why, the same thing they see in the Java, .NET, mobile web, etc. shops:

- Automated testing
- Test driven development
- Unit testing
- Code coverage
- Continuous integration

And a number of other concepts common to the Millennial developers coming out of school and submitting job applications to your Millennial boss.

“But MultiValue can’t do that!”

Really? I’d suggest you go to <http://intl-spectrum.com/magazine> and check out the articles on those subjects in the 2014 and 2015 issues. Yes it can. And it has been done very successfully.

And don’t forget about project management and development methodolo-

gies. I’m talking Agile, Scrum, Kanban, etc.

Now if you are looking for all of this to be available at no charge from the *database* vendor, I’m afraid you are rather out of touch. Nothing we have talked about in this series is exclusive to the MultiValue development environment (a mature product) or the database itself (mature and well proven). It’s all about learning new things that *tie into* the MultiValue database. Or new methods that *also apply* to the MultiValue environment.

It is all there now. Some of it has been for years.

So what are *you* going to do to attract new blood to the MultiValue community? **IS**

Is Your Membership Profile Up to Date?

International Spectrum is currently working on updating membership information for all our readers. We always like to have the most current information so that we can keep you informed of current news in the MultiValue Market.

Please go online and update your membership profile today!

intl-spectrum.com/membership