$7.00 U.S.

INTERNATIONAL

Spectrum®

THE MULTIVALUE TECHNOLOGY MAGAZINE | JULY/AUGUST 2013

Secure Coding for MultiValue

ALSO IN THIS ISSUE:

- Epic Fantasy in Business Tech
- Automated Integration Tests
- Mobile Apps with O4W — Part 3

intl-spectrum.com

GET CONNECTED.

Knowledge and Education for the MultiValue Professional.

About Our Professional Memberships

We are all busy in our day-to-day work and staying up-to-date with the current MultiValue technologies can be difficult.

Professional Memberships provide you access to knowledge, solutions, information, and code that you won't find in other locations.

Professional Membership Includes:

- Magazine in Electronic and Print
- Newsletter
- On-Demand Videos
- Live Webinars
- Discounted Conference Rates
- Research papers
- Case Studies
- Source Code

INTERNATIONAL Spectrum®

http://www.intl-spectrum.com/membership/

INTERNATIONAL Spectrum
THE MULTIVALUE TECHNOLOGY MAGAZINE

FEATURES | JULY/AUGUST 2013

6 **Secure Coding for MultiValue** Closing firewalls, email filtering, passwords, encryption, SSL, injection attacks — welcome to running with the big dogs. As MultiValue systems move into mainstream environments and become part of the enterprise's multi-platform, customer facing business solutions, we find ourselves having to deal with and defend against the same types of attacks our non-MultiValue partners have been plagued with. While there is a lot of literature about these issues available, unfortunately the MultiValue platform is not usually the topic of these discussions. This article is a start in closing that gap. **BY SUSAN JOSLYN**

10 **Business Tech: Epic Fantasy** There is little doubt in anyone's mind that our economic landscape is changing. This, of course, will also impact our use and implementation of information technologies. This issue's Business Tech takes a look at some of these changes and makes some interesting (perhaps even controversial) predictions. **BY CHARLES BAROUCH**

14 **Automated Testing Part 3: Integration Testing** Part 1 of this series explained how "mainstream" technologies develop and use automated testing and makes the case for pulling MultiValue testing methodologies into the 21st century if we are to continue to be seen as a viable application development platform. Part 2 tackled the topic of the Unit Test and how we might implement it in a well structured MultiValue solution. This article moves on to the topic of Integration Testing — do all the pieces fit together and work properly? **BY BRIAN LEACH**

18 **Building A Mobile Application With O4W – Part 3** Part 1 demonstrated how to build a simple display form using the O4W APIs Part 2 took that same routine and enhanced it to provide more interactivity, user input, and advanced interface elements. The concluding Part 3 shows how to process the entered user data. **BY BRYAN SHUMSKY, REVELATION SOFTWARE**

DEPARTMENTS

From the Inside page 4
From the Press Room page 12
Clif Notes page 23

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

It is the time of year that I start planning for the next Spectrum Conference. If you haven't seen the ad for the 2014 Conference, it will be Apr 7th-10th in Phoenix, AZ. We are having it at the same venue as the 2013 conference — The Wigwam Resort.

While it seems like the conference is pretty far away, it really isn't. I have to start planning my sessions so you have something to sell to your bosses. I get great content from the industry experts that provide sessions for me every year, but I would like to know what you as a MultiValue user and developer would like to see.

The demands on your IT department are getting more specific, which leads to specific examples or business solutions that you need to know. I remember those days quite well, and I watch the trends that are coming to see which ones our industry are likely to need to be educated on.

The Spectrum Conference is designed for you,the User. I want to make sure you get the materials, education, and experiences that are needed to maximize your ability to create efficient enterprise applications.

There are a lot of different trends floating around right now. If you are unsure what you would like to see, here are a few that I've noticed:

- Virtualization/Cloud Computing

  Everyone is talking Cloud Computing, and Virtualization, and for pretty good reason. What kind of information are you looking for? Server virtualization, planning, cloud API options?

- Tablets and Cell Phone Apps

  This topic is a pretty sure thing for 2014, but I would like to hear your input. What are your company demands: Native Apps, Web Apps, iPad, Android, BlackBerry?

- Windows 7/Windows 8/Metro

  If you haven't worked with Windows 8 much yet, it can be a pain. Would you like to see general sessions on how to use Windows 8? Want to see what Windows 8 brings to the enterprise? These general sessions would not likely be MultiValue specific, but sometimes we need to know about other software in order to make our job easier.

- Integration — Email, CRM, EMR, LDAP, Active Directory

  Do you have other applications within your environment that you need to integrate with. We have been an island all to ourselves for so long, we forget that our business has information other systems can use or need for business functions. What integration options do you need?

- External Programing Languages

  MultiValue Basic has served us well, but we need to interact with other systems, so that means other programing environments. What languages does your business require you to work with? .NET, Java, Python, Javascript, HTML, JSON... there are many others.

These are just a few examples, but Iwould like to hear from you to see what is most interesting or trends and topics that you would like to have available.

Email me: nathan@intl-spectrum.com

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

INTERNATIONAL Spectrum
JULY/AUGUST 2013

NATHAN RECTOR
President

CLIFTON OLIVER
Editor

TRACEY RECTOR
Layout

Learn more about the MultiValue Symbol and see what MulitValue 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 Specturm, Inc., 8956 Fox Drive #102, Thornton, CO 80260; Tel: 720/259-1356; Fax: 603/250-0664 E-Mail: request@intl-spectrum.com. Copyright 2013 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.

REVELATION
SOFTWARE

Your database development DESTINY has yet to be written

Invent the future with OpenInsight

Software architects need database development tools that evolve with their rapidly changing business landscape. We are Revelation Software, creators of the OpenInsight Development Suite, bringing you one of the best browser-based, mobile computing and robust reporting toolkits on the market. Go to revelation.com and start inventing your next great software solution today.

OpenInsight 9.4

Secure Coding for MultiValue

BY SUSAN JOSLYN

Remember when you were a kid and all you wanted was to be a "grown up?" Everyone told you not to rush it. "It's not as great as you think it will be," they would say, knowingly. The situation is not so different for those of us in the MultiValue community. We have been talking for years about wanting to be more mainstream and more recognized, but as we extend our reach we may find that, just like being a grown-up, being mainstream isn't all it is cracked up to be. This is particularly true as it relates to security. For years we've been able to crouch behind "security by obscurity" as cover. Unfortunately, even if our flexible database remains fairly obscure, we are all busy pushing our data out to SQL and widely used reporting databases and web interfaces. We are moving stuff around through very mainstream methods — and lauding the fact that we are doing so. User interfaces and networking and generally data in motion are NOT unique to the MultiValue world, so we must consider all of the usual security issues that any IT entity must consider — firewalls, email filtering, passwords, encryption, and SSL. In addition all of that, there really are very specific ways that data and security breaches can be orchestrated against our own beloved database and programming environments.

My own perspective on this issue took a right turn several years ago when I was speaking to one of my typically small but fierce audiences of security enthusiasts. When I mentioned a very common SQL injection threat, Dan McGrath (now Managing Director U2 Servers Lab) began describing how a similar attack could be implemented using a Basic program and our own retrieval language. Dan was working with data that had to be secure and was doing some creative thinking about hacking into and hacking up MultiValue systems that most of us weren't. Some of this discussion came from ideas born in his devious mind. More information and some fascinating yet terrifying examples can be found at his blog http://u2tech.wordpress.com/.

The best way to get the coders, the testers, and the users involved and aware of security issues is to begin by at least mentioning security when defining the requirement.

There are some general security issues and ideas that are the same no matter what the platform, the threat, or the year. The details between these guideposts are the specific vulnerabilities in MultiValue and secure coding practices for the environment. To keep it all in perspective, we'll start from the top.

Awareness — Yours and Your Users'

Inarguably the best defense against any threat or attack is awareness — both within the IT community and within the larger user community. Make sure users are aware of what could happen if they clicked a link and the types of persuasion they may run into. Make a point to understand the motivation — what would someone be after, and why — and convey that to the user base. This way they can think creatively when confronted with a new angle. For example, a rising awareness issue is employment ads. When the company advertises for specific skills they are stating very publicly exactly what technology is in play. Can you hire people without describing the skills? Of course not. But risks can be mitigated with some thought to wording, so HR must not be excluded from the company-wide awareness initiative!

Threat Modeling & Defense in Depth

In order to keep users aware of the nature of new threats, and to maintain a state of vigilance in general, IT must affect good security controls. Threat Modeling helps in understanding the risk — the vulnerabilities and the threats. Defense in depth refers to having a multi-faceted approach to security. It's like having a lock on the door and a deadbolt and a chain. And maybe a barbed wire fence. And a big dog. Probably not a moat. Some specific security approaches do go out of favor.

Reducing the Attack Surface

This is the hip new way to say "minimize access." The "attack surface" is any point of access into the system or the data. This encompasses issues around data at rest and in motion, firewalls, VPN, and other access. These concerns are common to all IT enterprises regardless of platform or industry. The fundamental human issues related to access are 1) passwords: don't use default passwords or allow static passwords; 2) the

Digital Version On-Line

Scan Me!

principle of least privilege: don't allow any individual access to anything unless it is necessary; and 3) defined roles / segregation of duties: making sure that other eyes/sign-offs are involved in any critical activities. These three can be addressed in the code itself under the duel umbrellas of IT General Controls (an example of which is passwords) and Application Controls (defining and enforcing who can do what within the application software). But the real action is in sanitizing data, including validating input from human or non-human sources.

Security in All Stages of the Software Lifecycle

Requirements Phase

The best way to get the coders, the testers, and the users involved and aware of security issues is to begin by at least *mentioning* security when defining the requirement. It can be as simple as remembering to pose the question, "Have the security implications of this new feature been considered?"

Coding Phase

As programmers, our focus is on providing quality solutions to our users by thinking about the "user friendliness" of what we are developing. We don't want to lose that, but we may need to temper it a bit by thinking about whether we are developing applications that are "too friendly." We must ask ourselves, "Have we put too much power in the hands of an unhappy or unwelcome user?" And while we don't want to be paranoid, neither do we want to be too trusting of data that comes into the system from outside sources. Or even inside MV data that comes through third-party applications. Unlike Stella from *A Streetcar Named Desire*, we mustn't depend upon the kindness of strangers when it comes to sanitizing incoming data.

When we are coding and reviewing code here are some general risks with a MultiValue twist that we should consider:

Math: Potential integer arithmetic issues. Equations that are incorrect can corrupt data (precision loss/rounding errors) or bring the system to its knees due to overflows.

Input: When something can get into the system from the tip of human fingers it should be carefully examined before it is welcomed in to the rest of your data where it may do harm. It is not just a convenience to the user to remind them of the sort of data they should be entering, but by making certain the data is what was expected we can protect the system from accidental or intentional corruption. When input data should be a number, check for a number. When it should be a date, check for a date. Not just any date or any number, either. Make sure it is in a reasonable range. Range checks are rather infrequently found in MultiValue applications, but extremely large or incorrectly small numbers or massive ranges of dates are a very easy way to force a breakdown.

In addition to testing for range we may want to go further and rely more on the practice of testing input against a white list. In non-security terminology that might be a code file or a reference table. Whatever you call it, it is a list of valid inputs so that something that comes in from an outside source won't make it past the fence if it isn't specifically listed as valid.

Input: Executes The riskiest thing we ever do in a MultiValue Basic program is an EXECUTE. This is because we can execute almost anything. Many applications cleverly build up executable statements based on input from the user. This opens the door for a creative vandal. In the same way that a SQL injection can cause a dump of a whole database, allowing untested strings to be built into executes can wreak all kinds of havoc. Here are some things to keep in mind.

- The 'USING' keyword allows a statement to be created against one file, using the dictionary of another file. An intrepid hacker could create a dictionary that executes subroutines, and more, that would not be detected even in a carefully protected data file and its associated dictionary. Then simply add the phrase 'USING OTHERDICT' when the prompt asks for customer name. It's all about the quote marks.
- Quotes and Wildcards: If allowing a user to input their own criteria, sanitize the entry for extra quotes and wildcard characters. Imagine all the things that could be added to an execute by expanding it this way.

Input: Marker Injection Sanitize for marker/control characters — attribute marks, value marks. Most folks that have been around the MultiValue database for any length of time have dealt with a data file corrupted by control characters. Such a simple thing can cause mysterious problems.

Input: Keys Allowing external input to be written out as the key to a file is risky business. When this is necessary, be sure to carefully validate anything that may be used as the key to a file. The wrong character in a record key can corrupt an entire database. It can skew the hashing algorithm or throw off indexes, too — things that will not show up immediately but will degrade performance.

Input: external input as called subroutines Avoid using external input to set dynamic subroutine names for CALL @ statements without white-listing the input first. If the external input is compromised, it may enable un-intended subroutines to be called.

BASIC Coding vulnerabilities

- **Lock / write / release** matchup. A common accidental mistake, this can be readily capitalized on for nefarious purpose. When coding and when reviewing code, be sure to match up the reads and releases. If an item isn't written after it has been read-locked, it must be released. Otherwise a lock table can overflow and bring the system to its knees.
- **Dynamic arrays.** Similarly a dynamic array will work very well and very flexibly — even when used badly. It will try and try. But like the lane-line painter who leaves the can of paint at the end of the road, as the distance back to the paint increases the effort will slow down

Continues on page 8

SECURE CODING FOR MULTI-VALUE

Continued from page 7

accordingly, while the amount of effort required will continue to increase.

- **Oversharing.** Error messages and help prompts are tricky beasts on so many levels. Ideally they pose a training opportunity. And of course they are useless if they don't present enough of the right information to help the user. But beware of exposing too much information, especially about the underpinnings of the system. Too many specific details about the inner workings of the software can help a casual intruder.

Other MultiValue-specific security miscellany

- **TCL commands and verbs.** The availability of TCL commands that can compromise data is at the root of the cautions about building execute statements in Basic. While paying close attention to what users can enter that can be executed is still sound advice, the use of Remote VOCs can provide sweeping security by limiting what users or what processes can execute what commands against what files, even from TCL.
- **Spooler closing.** Another example of something that is often done by accident, not closing a spooler entry can create a monster of epic proportion. This is something that show be on the code-review check-list. It's a cheap hack.
- **Uniobjects.** This handy tool provided by U2 can really give developers a leg-up to providing various user interfaces. Unfortunately it can give an unfriendly a similar leg-up. For one thing it is a dangerous security hole to have a port open to anything except internal servers. Then, Uniobjects has no fine-grained server-side control of what actions can be done, or commands issued. In the default configuration, as long as you can log in, you can get a free pass to the back-end data. There are some steps that will help, though. One is to create a UOLOGIN subroutine to whitelist users who can make database calls via UniObjects. (Particularly useful when you use middle-tier services, as you only need to allow the services usernames to have access.) As mentioned above, Remote VOCs can be implemented to protect verbs through the use of white lists and Access Control Lists.

Hashed Files

The hashing and group structure of a MultiValue database is ripe with opportunity for a well-informed admin with a grudge. Loading up a file beyond its reasonable capacity, skewing it with certain alpha or numeric key structures, and other strategies can bring a particular file or even the entire system to its knees. This gets back to sanitizing input — particularly whatever will be the key to the file. Be sure to test it for the right characters and range and general reasonable-ness.

Testing Phase

Code Review / Peer Review. Happily, it is more common than in years past for companies to have a policy of peer reviewing or even to have dedicated code reviewers. Either way, this is a really good idea. Having another pair of eyes looking over any new or changed software before it goes for user testing — and especially for that review to be conducted with security in mind — can up the game considerably. Bear in mind that the critical point here is that it must not be the programmer reviewing his or her own work. The idea of a code review is to have someone look at it fresh, to see it differently. Another smart idea is for the security folks to build a list of things to consider during the code review. This would be specifically for security related issues,separate from whatever other things your company may wish to review such as coding standards, etc.

Failing gracefully. The first item on this new list of security considerations during a technical review is to force a failure and see how it behaves — see if any gaping holes are opened in the security of the system. Does a failure drop the user to TCL? Or into a powerful debugger?

There are a couple of settings in a U2 system that can mitigate the risks that go along with a program fail. On Unidata there is an option 41 which, when on, will loop back to the program when an "execute" fails, rather than dropping out to TCL (see also UDT.OPTOINS 105). An entry called ON.ABORT in the VOC can direct control to another command, process, or program whenever the ctrl-break, debugger, or drop to TCL does occur.

Includes A peer reviewer/code reviewer would be well advised to take a look at any includes in a program — not just assume that they do what their name seems to imply. Imagine the havoc that can be wreaked by tweaking an include statement to equate common and database variables to something other than what it sounds like they are! Particular attention should be paid to security related COMMON and EQUATE statements.

Logs, Audits and Reports

Logs are helpful for securing, monitoring and debugging applications. They can be particularly helpful for logging the status of phantom processes. It is important however, that log files do not contain sensitive data that is not needed, such as passwords. Make sure you are not recording normally secure information in unsecured log files. Log files are likely to turn up on develop-

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

ers machines, in print-outs and generally unencrypted.

Don't log too little or useless info. Log files should contain enough information to determine when something happened, what happened exactly and by whom. Effective logging not only gives you diagnostic capabilities for when mistakes happen (user OR developer), but can also act as a deterrent for would-be malicious parties.

Don't log too much information. Big Data is a lot like the Boogie Man. An invisible threat that is scary in theory but so far we haven't run into it face to face. Unlike the Boogie Man, however, the threat of Big Data is going to remain — and become more real — as we become grown-ups. Not only are we dealing with large amounts of varied data formats, but we are also creating big data when we log access to the data! Our log files and audit trails quickly become useless when they are too many, too much, or too detailed. Next, we want to think about protecting the logs themselves. And logging access to the logs. Log logs. This can quickly get out of hand! Not only should content, format, security, and duration of the logs be considered when you are building new audit trails but we need to consider the audit trails left by spoolers, COMOs and external communications.

Security is big topic. Specific security for MultiValue is even a big topic. It is getting bigger and more urgent all the time. The most important piece of advice offered here is to remember that this list is incomplete no matter how complete it could be. We have to keep thinking about security in every new thing that we do. Talk about it, think about it, keep it in mind. IS

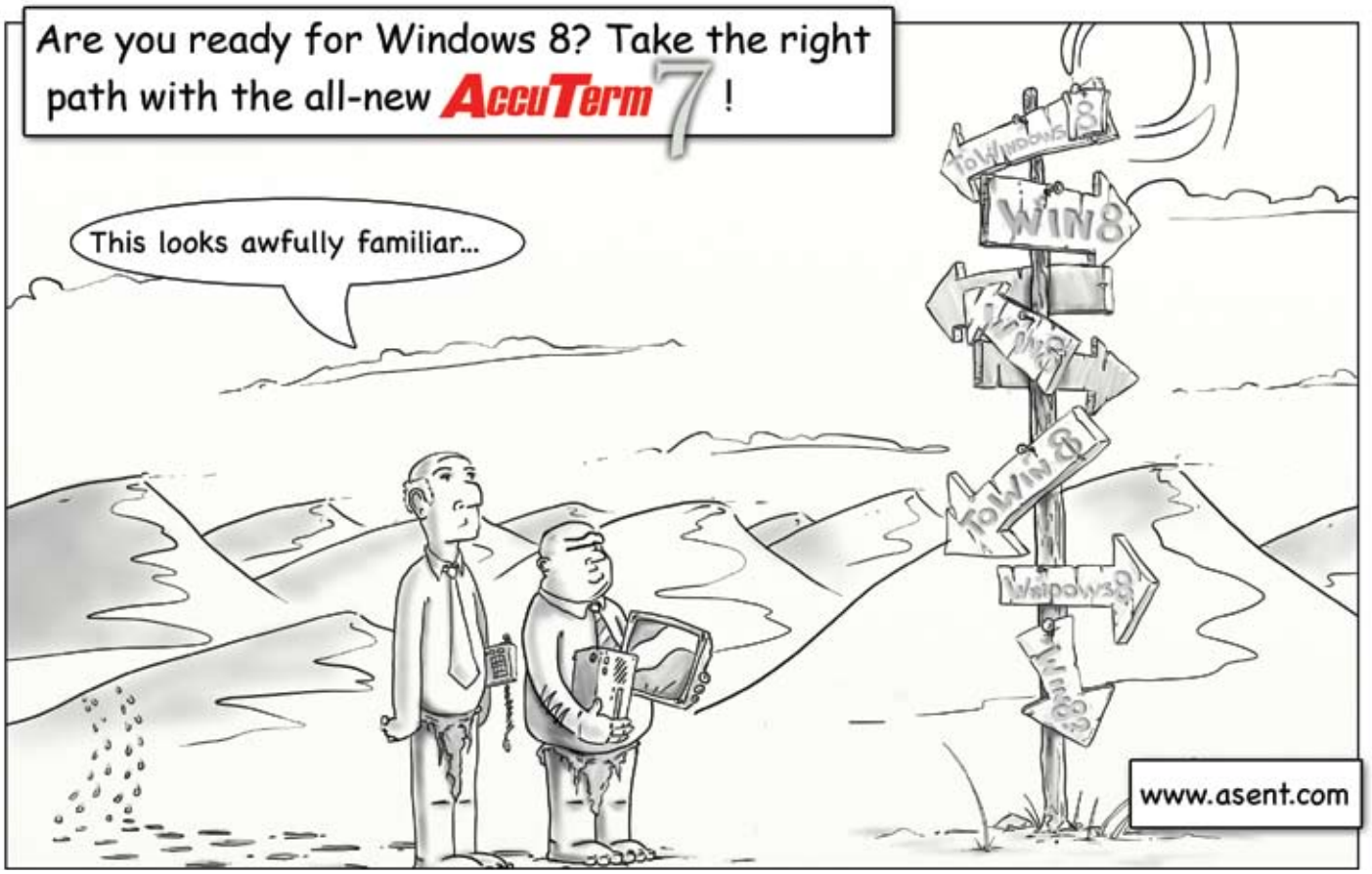
SUSAN JOSLYN is the President of SJ+ Systems Associates, Inc. and is the author of PRC?, a complete, integrated software development life-cycle management / IT Governance tool for U2. She has worked with U2 (nee Pick/Multivalue) and SB+ software the beginning (both hers and its) and has specialized in IT Governance, including quality, compliance and life-cycle productivity issues since the early 1990's.

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@eslny.com

Consultants Needed Nationwide



Epic Fantasy

BY CHARLES BAROUCH

The land was full of those who trusted the multitude of soothsayers, prognosticators, and fortune tellers. They felt that the future could be predicted; be known. Then the Euro nearly collapsed while the U.S. Dollar seemed to be in free fall. And no one seemed to know if it was safe to start hiring again.

We live in a time of broken words, of assurances from those supposedly in the know, that showed us how little any of us knew. The epic fantasy of the various 1980's business models has been broken over and over again. At the risk of joining the crowd of people who have guessed wrong, here's my predictions for the emerging economy.

Buckminster Fuller Was Right

And there were monsters in that time, each with an appetite for eating up the jobs which people needed. Finally, one man arose who was brave enough to tell the truth about the vanishing jobs. It took some time before the people were brave enough to believe.

"We should do away with the absolutely specious notion that everybody has to earn a living. It is a fact today that one in ten thousand of us can make a technological breakthrough capable of supporting all the rest. The youth of today are absolutely right in recognizing this nonsense of earning a living. We keep inventing jobs because of this false idea that everybody has to be employed at some kind of drudgery because, according to Malthusian Darwinian theory, he must justify his right to exist. So we have inspectors of inspectors and people making instruments for inspectors to inspect inspectors. The true business of people should be to go back to school and think about whatever it was they were thinking about before somebody came along and told them they had to earn a living." — Richard Buckminster Fuller

There are several monsters lurking in the dark corners of the business world that are making it impossible for us to have a job for everyone who wants/needs one.

There are several monsters lurking in the dark corners of the business world that are making it impossible for us to have a job for everyone who wants/needs one.

The biggest monster is mergers. One hundred small business means more senior — good pay — jobs than one mega-business. As the rules against anti-trust are being broken down globally, the pressure rises to break them down faster. After all, if France allows a monolithic company to rise, Germany has to have a behemoth ready to go toe-to-toe in the international markets. Once there are two, it only stands to reason that everyone needs one. The consequence is fewer jobs with a good wage on top of fewer jobs overall.

The next monster is efficiency. Fewer steps is more efficient. When you eliminate steps, you get the same results with less cumulative hours of work, i.e. less jobs.

Then comes the market forces: the stock market, the price sensitivity of your market, and the fear of shrinking margins. The biggest cost in most businesses is labor. How do you lower costs? Fewer employees.

Lastly, we are one of the monsters in the dark. We automate things which removed manual steps. We've already established that steps = hours and hours = jobs.

So, unless we make the leap away from capitalism — not going to happen in our lifetime — we are headed toward either an increasing percentage of permanently unemployed people or a massive downturn in population. Our role as monsters does not protect us from job loss. There are a lot of qualified tech people on the street already.

Employee-free Companies Will Become a Growing Segment

And the farm was empty of people, but the cows were still milked and the sheep were shorn. It was as if an army of ghosts had put aside their haunting and learned the value of industriousness.

I have two clients right now who have businesses with no employees. Everyone, owners included, are consultants and contrac-

Digital Version On-Line

Scan Me!

tors. I have others where every employee is a part-timer with a full time job doing something else. While these models work well for some businesses, I predict that their cost/benefit ratio is going to encourage people to try them more widely than they should.

Despite the certainty of notable failures, I see this concept dominating certain industries. For those who make it work, there are tremendous upsides. Having a standing in two or three businesses like this allows you to protect your income from downturns in any one venture. Important: make sure each business is in a different market segment.

Good business for this model include most creative ventures or collecting. Think antiques, art, comic books, and the like, when you look at this category. Business software might also be a good fit, depending on the circumstances. I've found it a useful model in the chaotic publishing world.

Chaos is Good for Small Businesses

The world had gone all atilt and only the fleet of foot could scramble to the places where there was safe footing.

Speaking if chaos… The generally accepted wisdom that small businesses are more agile is largely correct. In a growing market, agility gradually becomes a fringe skill. In a failing market, or a market in upheaval, agility becomes critical. If you have an entrepreneurial spirit, you may want to "run toward the fire" and start a business in a market that everyone else is abandoning.

Business Technologists, like us, have an edge here. We know how to spin up flexible infrastructure and how to keep it from getting rigid. The hard part is recognizing when the market starts to mature. That's when you have to start switching gears and begin focus on stability over flexibility.

Giants Fall Slowly

While some found shelter from the storm, the biggest ones could not fit entirely and parts and bits of them were exposed to the full wrath of the whipping winds of change.

I think we are going to see a lot of giant companies decline over the next few years but they, for the most part, will drift down instead of plummeting. It used to be understood that the bigger they are, the harder they fall, but that wisdom is not as solid as it used to be. In the tech world, expect everyone to be in some sort of slow motion spiral in the next few years. If they aren't, they probably have significant revenue from semi-tech or non-tech components.

Advertising is Headed for Trouble

The heralds and jesters were used to a protection due their special roles in society, but even their safety was uncertain in this strange new world.

The ad world is based on getting the word out to customers. While the job market will rebound a little, the net percentage of the population with money to spend is shrinking. Fewer people worth targeting means a sea-change in how marketing works. The rise of the social network is also re-writing the Ad world's rules. Throw in the successes, and failures, of algorithmic marketing (think Google, Amazon, Facebook, for examples) and the implosions will only get worse. If there is any place where giants might fall faster, this would be a prime example.

Behind it all: The Dirty Secret of Capitalism, and Therefore, of Business

Pretend I pass a bad check for fifteen thousand dollars to Bob. My bank honors it by mistake. He spends half the money clearing his debt with Penny and the other half he invests in Sally's business. Penny owes Victor, so she clears a portion of her debt. Sally and Victor are customers of mine who can now settle up their bills. Stay with me, we are nearing the finish line.

The bank finds their error and demands that I cover the check. I take the money I got from Victor and Sally and make good. So, the check was bad but: (1) I owe Bob nothing, (2) Bob owes Penny nothing, (3) Bob owes Sally nothing, (4) Penny, Sally, and Victor EACH cleared over seven thousand dollars of her debt, (5) I am even with the bank, and (6) everyone, including the bank, has exactly the same amount of money they started with before I accidentally fixed the economy by passing a bad check.

Despite how fictional it all is, the people at each end of the economy, the rich and poor, feel real-world effects. What does that mean? Economies are fiction, but poverty and wealth are real. That's the dirty little secret of capitalism. IS

CHARLES BAROUCH is the CTO of HDWP, Inc. He can be contacted at www.hdwp.com

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

prc®

Announcing U2 Toolkit for .NET (U2NETDK) v1.2 - .NET capabilities for access to powerful U2 databases

If you need to build applications quickly with the look and feel of Microsoft .NET framework while working with the UniData or UniVerse databases, U2 Toolkit for .NET (U2NETDK) is the tool for you.

U2NETDK integrates updated UniObjects for .NET (UO.NET) native API, speeds development with U2 Database Add-Ins for Visual Studio, and interfaces with Microsoft standards SSIS and SSRS. The U2 Database Add-Ins for Visual Studio are a collection of features that integrate seamlessly into your Visual Studio development environment so that you can work with U2 objects such as tables, views, and subroutines.

SSIS extends capabilities associated with data migration such as copying files to update your data warehouse, as well as extract, transform, and load data into or from your U2 database. Interfacing with SSRS allows you to rapidly create and deploy new reports.

With this release developers can now treat data as logical groupings (entities) and focus on business logic. Microsoft's Entity Data Model (EDM) maps these entities to a physical UniData or UniVerse database and allows the developer to manipulate this data at the entity level. Developers can now use declarative queries and execute updates over entities and entity relationships. ■

Cairo Amman Bank goes live with jBASE technology in Jordan's largest ever core banking roll-out

Temenos, the market leading provider of mission critical software to the financial services industry, announced that Cairo Amman Bank, a full service bank with 106 branches in Jordan and Palestine, has gone live with T24, Temenos' award-winning core banking system based on jBASE technology. The deployment of T24 replaces the bank's legacy core banking system and consolidates multiple banking modules. Through centralizing its operations, Cairo Amman Bank expects to see immediate improvements in operational efficiency including automation of Real Time Gross Settlement payments, freeing up a team of people to support additional growth in the business and laying the infrastructure to support sustained growth in income and profitability.

June 2012 saw Cairo Amman Bank's Abu Nseir branch go-live as a pilot branch, followed by Treasury department operations in October 2012. The Zara Mall branch underwent the implementation of a model branch in March 2013 and the remaining 84 branches in Jordan and 21 in Palestine are scheduled to go live by the end of 2013. As part of the move to T24, the bank is also installing a Temenos' anti-money laundering (AML) screening solution.

Cairo Amman Bank selected T24 based upon Temenos' deep understanding of the local banking market and its proven track record of implementation of modern technological IT banking modules and support systems.

Kamal Al-Bakri, General Manager, Cairo Amman Bank says: "Temenos has delivered Cairo Amman Bank a modern, best-in-class core system, pivotal to the future success of the bank. We have invested in T24 to centralize and unify our core platform, eliminating complexity and lowering total cost of ownership. With T24 we now have an open and modern platform that integrates and interfaces with any external system and has the flexibility and scalability to grow with us. We found in Temenos a like-minded, innovative partner, harnessing technology to provide a truly differentiating service."

According to Mr. Clive Ketteridge, General Manager of jBASE at TEMENOS, the underlying run-time and database of the bank's implementation is jBASE. "One of the prerequisites for TEMENOS' T24 banking application is the jBASE open connectivity technology integrated with the banking functionality and coupled with an open database such as Oracle, IBM's DB2, Microsoft's SQL Server or jBASE, which provides clients with numerous options with regard to future direction. This undoubtedly offers the stability Cairo Amman Bank was looking for in rolling out their solution. Our customers complement us often on the ease of deployment and ongoing housekeeping that jBASE affords."

For more information, visit www.jBASE.com.

About jBASE International

jBASE International, a member of the Mpower1 group of MultiValue companies, is a leading supplier of database management software and web enabling tools for developing, deploying, and maintaining business applications solutions. The flagship product, jBASE, was designed from the ground up to be an open database product that would bring the strengths of MultiValue technology into the mainstream computing market. With exclusive worldwide distribution rights, jBASE International offers technologies and assistance that allow businesses to thrive into the future.

About Temenos

Founded in 1993 and listed on the Swiss Stock Exchange (SIX: TEMN), Temenos Group AG is the market leading provider of

banking software systems to retail, corporate, universal, private, Islamic, microfinance and community banks, wealth managers, and other financial institutions. Headquartered in Geneva with more than 59 offices worldwide, Temenos software is proven in over 1,500 customer deployments in more than 125 countries across the world. Temenos' products provide advanced technology and rich functionality, incorporating best practice processes that leverage Temenos' expertise around the globe. Temenos customers are proven to be more profitable than their peers: in the period 2008-2010, Temenos customers enjoyed on average a 30% higher return on assets, a 46% higher return on capital and an 8.5 percentage point lower cost/income ratio than banks running legacy applications. For more information please visit www.temenos.com. ■

Pick Cloud, Inc. Announces Reseller Agreement With Synergetic Data Systems,Inc.

Pick Cloud, Inc., a provider of fully managed hosting solutions for the MultiValue market, today announced they have signed a reseller agreement with Synergetic Data Systems, Inc. (SDSI) to integrate CirrusPrint, SDSI's cloud and network printing solution. "We are very excited to be working with SDSI to provide a cloud printing solution to our MultiValue clients, says Mark Pick, CEO of Pick Cloud Inc. Now our customers can enjoy seamless, secure and simple printing from the cloud without the necessary hardware overhead associated with a virtual private network. This is yet another great addition to our cloud suite of products that makes moving to the cloud even easier."

The reseller agreement enables Pick Cloud, Inc. to offer and leverage CirrusPrint 's many capabilities. CirrusPrint runs on Windows, Linux and many Unix platforms. It streamlines and manages remote printing and documents transfer between the cloud and remote computer systems and local printers and file storage systems. Offering many features for reducing the cost of printing, it also improves document processes and simplifies printer management. Secure printing is also available via SSL.

"SDSI is a leader in the print management, document delivery and document management marketplace. Our CirrusPrint product is a great example of that," says John R. Wilson, SDSI's Vice President and Director of Sales and Marketing. "We are pleased to be working with Pick Cloud to provide a cloud printing solution to their MultiValue clients. We strongly believe that CirrusPrint can add tremendous value to MultiValue customers who want to streamline and manage their cloud printing process."

About Pick Cloud, Inc.

Pick Cloud Inc., located in Newport Beach, CA and founded by Mark Pick (son of Dick Pick), provides a cost effective, worry free, secure, environment while at the same time preserving and rejuvenating MultiValue applications through smarter processes and more efficient technology. More information about Pick Cloud, Inc.'s products and services can be found at www.mypickcloud.com.

About Synergetic Data Systems, Inc.

SDSI markets CirrusPrint and other software tools through an international network of resellers, developers and integrators. Other SDSI products include the UnForm Document Management Solution,General Report Writer, sdOffice, dServe, and MailCall. Visit SDSI's website at synergeticdata.com for more information about the company and its products. For more information about CirrusPrint, please visit www.cirrusprint.com. ■

Kourier Integrator - Release 4 Announced

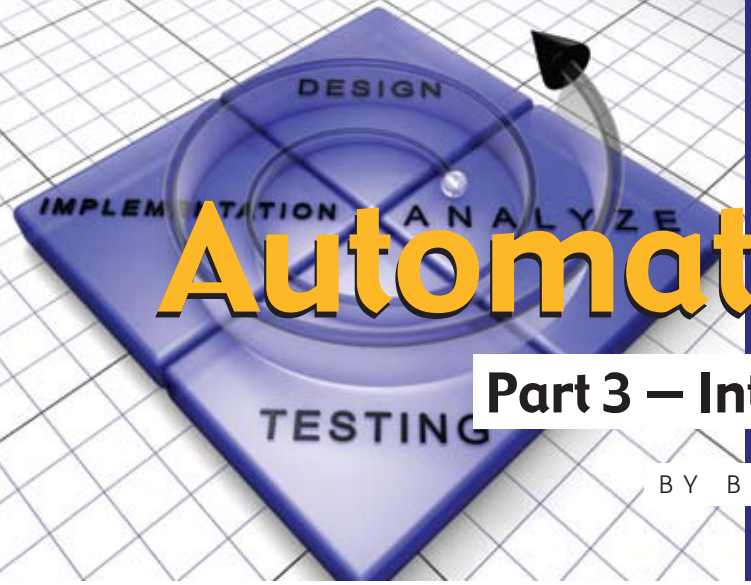
Kore Technologies is pleased to announce the availability of Kourier Integrator - Release 4.

Kourier Integrator is Kore's enterprise integration suite providing both Extract, Transform, and Load (ETL) and Enterprise Application Integration (EAI) capabilities for connecting Rocket Software UniData/UniVerse (U2) applications to Microsoft SQL Server and other best-in-class applications.

"Increased Connectivity" is the theme for this new release, which focuses on expanding the ability to connect with the multiple data sources that are typically used in companies today. Release 4 includes two major enhancements:

- Seamlessly integrate and consolidate information from U2 and/or other data sources (e.g., Oracle, MySQL, Microsoft Access, and text files) into a Microsoft SQL Server database. This gives companies the ability to manage data from their islands of automation using a single ETL/EAI solution.
- Native support for Microsoft SQL Server 2012 and Integration Services (SSIS). This is important for companies that want to leverage the latest database technology from Microsoft, and take advantage of the additional performance that will help them move large amounts of data quickly.

Learn more about how you can quickly create a Microsoft SQL Server 2012 data warehouse with information from U2 and/or other data sources in your organization using Kourier Integrator, please contact Kore Technologies at 866-763-5673 or send an email to sales@koretech.com. ■



Automated Testing

Part 3 — Integration Testing

BY BRIAN LEACH

In the previous article I introduced the subjects of unit testing and test driven development. In this next piece we move on to the next phase in automated testing — integration testing. By now you have completed your lovingly crafted and fully test covered new code unit. Before you release it into the wild, you need to make sure that it will play nicely with the other features.

Integration testing primarily seeks to ensure that:

- New or amended code does not disrupt the code base.
- The code operates in context.
- The code can be safely deployed.

Code Disruption

In his excellent book *Joel on Software*, Joel Spolsky describes the Daily Build undertaken when he was in charge of Excel development at Microsoft. The purpose was to ensure that all necessary changes were checked in and functional at the end of the day, and the punishment for breaking the build was the shame and rightful opprobrium from ones' peers and the penalty of babysitting the next builds (until the next culprit took over). There was a serious side to this — at any point they knew they could release a build. Agile practices take a similar tack today.

Integration testing ideally takes place on a dedicated SIT or a like-live system that allows you to ensure that the code works in context after it has been deployed. Integration testing is as much about testing the deployment as the build.

In the MultiValue world we do not have large executables linked from multiple sources, in the way of a Java or C# project. Nor do we have any metadata that can be examined to ensure that libraries will expose the right method signatures. Instead, we build separate units in the forms of programs, subroutines, external functions, GCI routines, PROCs, paragraphs, and other assets that cooperate to form a solution. These are only discovered at run time, and so the only ways to ensure that the system will remain undamaged following a code release are through a mixture of static code analysis and exercising the system.

Static code analysis can take you part of the way. In mvTest today, and before as part of a pre-compiler at my previous work, I include a simple code checker that can spot obvious anomalies: mismatches in the number of arguments on a subroutine call or in declared sizes of common blocks, subroutines that cannot be found, and warnings over variable name transpositions and casing that might not be caught by the compiler. But there are limits on what a code check can reasonably deliver, since indirect subroutine calls (Call @Name), passed file variables, and various dynamic calls (creating and opening Q pointers, generating selection statements etc.) can only really be caught at runtime. So a code checker is really a handy hint, a first port of call, and not something to rely upon. Exercising the system through workflow testing is the only safe solution.

Coding Context

The focus of unit testing is deliberately kept very narrow to ensure tight red/green/refactor cycles that can be quickly repeated during active development and test the low level behaviour out of context ("build the code right"). Once the code has passed its unit testing, we can widen our gaze to include those workflows and scenarios in which the code is destined to play it part ("build the right code"). Unit tests can typically be executed in any order but for integration testing we need to think in terms of sequences.

A typical example of such a work flow can be found in a support system website I recently developed for a software house. This allowed end users to log calls and to have them approved or rejected by their supervisors before being passed to the sup-

Digital Version On-Line

Scan Me!

port team. Having developed the back end of the application as a series of UniVerse Basic subroutines, each with its own unit test script, the next stage involved sketching these out into a number of workflows, for example:

- User logs a request -> request rejected -> email sent -> request removed.
- User logs request -> request accepted -> email sent -> support notified -> diary updated.
- Additional information requested -> user supplies information -> support notified.
- Additional information requested -> user uploads attachment -> support notified.
- User searches for request -> request retrieved -> user closes call -> support notified.

You may notice that these are very similar to acceptance criteria and document how the system should operate in the same way

Continues on page 16

AUTOMATED TESTING PART 3: INTEGRATION TESTING

Continued from page 15

that unit tests should document how the code works.

Most testing frameworks have some concept of a sequenced test, whether scripted or recorded from user activities. In Visual Studio you can create an Ordered Test that batches together a set of tests in a defined order to simulate a workflow. More sophisticated automated acceptance testing tools like FitNesse can match a test to an acceptance specification. For mvTest I discovered that I needed to take the sequence concept a stage further to allow for the volume and concurrency testing need to exercise a MultiValue application, and so the individual tests can be arranged into batches that operate in sequence, and those batches can then be arranged into test runs. Test runs can be kicked off in parallel with the batches executed in a randomized order for a set number of iterations.

Testing for a retail group approaching the New Year sales, the test plan involved creating a minimum number of orders, product lookups, deliveries, returns and so forth during each two hour slot that represented (and far exceeded) the peak operations of their stores, all by driving the application simulating user activity through workflows. To make it appear more realistic, natural timings for entering data were added to replace the normal high speed automated operations.

This is where a scripted testing language is important rather than a simple rig that pushes data and expects responses: scripting allows the tests to expect and handle exceptions such as locking or thresholds: if you are pounding a sales application through a test rig, you should expect to run out of stock or delivery slots at some point and for both your system and the test scripts to handle this appropriately.

Code Deployment

There is no point in testing your code on a development system only to watch it to break once it has been released. Integration testing ideally takes place on a dedicated SIT or a like-live system that allows you to ensure that the code works in context after it has been deployed. Integration testing is as much about testing the deployment as the build.

A prerequisite for any integration testing must be a smooth, predictable, and completely reproducible deployment model that requires little to no manual intervention. If your deployment requires you to create or copy assets by hand, you cannot predictably ensure that your deployment to live will succeed and there is no point in considering integration testing until that has been resolved. In this context, code involves the widest definition of source — not only your programs but every asset required to ensure the running of your application.

Here is a simple question you can ask yourself (quietly, with a drink close to hand). If you recompiled all your code tomorrow are you confident it would still work the same? If you lost all your dictionaries, could you simply reinstate them? If you redeployed all your setups in order, would you have a working solution?

Deployment solutions for MultiValue applications are few, but they do exist. mvInstaller generates packages similar in concept to the msi packages used for Windows setups, and includes code, data and commands to execute to complete the installation (fig. 1). Tellingly, it was the very first thing I wrote on becoming self-employed, as I needed to be certain I could deploy subsequent solutions successfully. Alternatives you should consider are Susan Joslyn's highly regarded lifecycle management product PRC, a well respected tool that covers deployment, auditing, and compliance. And Rocket is adding U2 deployment to their Aldon product.

Continuous Integration (CI)

Integration testing is a broad term that is variously applied to aspect of smoke testing, workflow testing, systems testing through to automated acceptance testing. What they all have in common is an aim to complete full code coverage in a hands-off manner, to complement user acceptance testing. Unfortunately, most of the leading MultiValue platforms cannot return code coverage statistics despite being implemented as run machines, a fact that severely restricts moves to ascertain the effectiveness of any test plans.

An extreme form of coverage is Continuous Integration (CI) testing following on from the daily or continuous builds mentioned above. In a CI environment, changes are automatically deployed to a black box server that runs through the integration tests

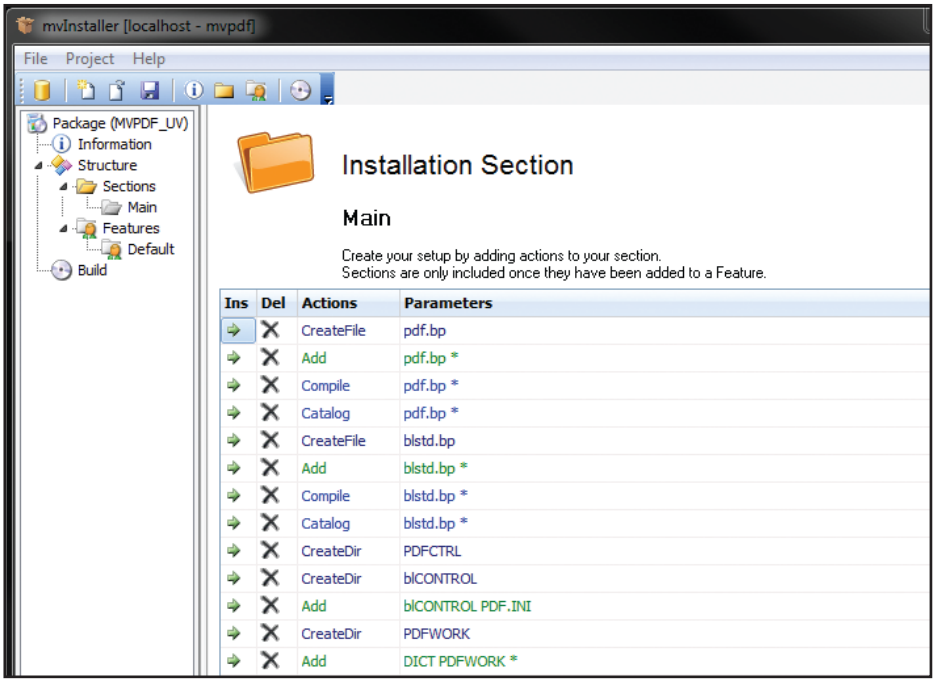


Fig. 1

on either a continual basis or automatically triggered following each new deployment. A CI server notifies the development team of any errors but stays quiet about successes, so that developers can focus on fixing, backing out, and redeploying.

UI Driven Testing

In the support system scenarios above I am testing the backend logic. By designing the application so that there is a clear separation between the UI and the business logic I can easily test these in isolation to localize any errors. But for legacy systems and those built using some 4GLs, that is not always possible.

A key decision in the design of mvTest was therefore to allow UI testing alongside the regular scripting and to have this operate as part of the same scripts. So the test runners include a virtual terminal emulator that can open Telnet, SSL, or SSH connections to the database and run the application in the same way as a terminal user. This would be possible using other standard telnet scripting tools that can also check the status of the screen and report on what is being displayed, but at the same time I wanted to be free to handle any error messages or lock conditions and to directly access the database through a regular connection to set up the initial data or to check what those screens have actually performed at the database level. That meant running the UI testing through a tool that was itself MultiValue-aware.

Most MultiValue testing is founded on some form of data entry, enquiry, or data flow, and so it was also important to add support for randomly creating or selecting test data: a date between two values, a key from another file, a line from an Excel spreadsheet.

So a typical example might go like this: run an order entry screen over Telnet, select a customer by lookup, enter a number of parts and quantities, check the totals displayed, file the screen, grab the order number reported, then from the other connection read that order from the database and check it has all the required data (fig. 2).

But what of other interfaces and UIs? Integration testing should ideally be end-to-end, exercising product feeds, web and desktop UIs and all the paraphernalia that attends a modern MultiValue solution. How far that is possible depends to a large extent on the mix (and cost) of technologies and the involvement of teams responsible for those, and the availability of appropriate testing tools to target those platforms.

Most environments today support open source testing tools such as Project White for Windows Forms and WPF, and WatiN or Selenium for web browser applications. All of these work through the UI and so run at a step removed from the database to automate testing through their respective target UIs. This limits their reach to those things that can be visible or instigated through the UI but with that limitation understood should form part of your arsenal.

Data feeds, message queues, and third party participants may always lie outside the practical scope of your integration testing and will thus need to be simulated unless you can dive directly into the code behind them. As with all such matters, how easily that can be done comes down to your application design, how far you have separated out the dependencies from the code business logic, and how willing you are to refactor your legacy code if not.

In the next article I will be looking at refactoring and some of the lessons learned in applying automated testing to legacy code. IS

BRIAN LEACH in an independent MultiValue consultant based in the UK. Brian produces a wide range of tools and training resources aimed at MultiValue developers. Learn more at http://www.brianleach.co.uk.

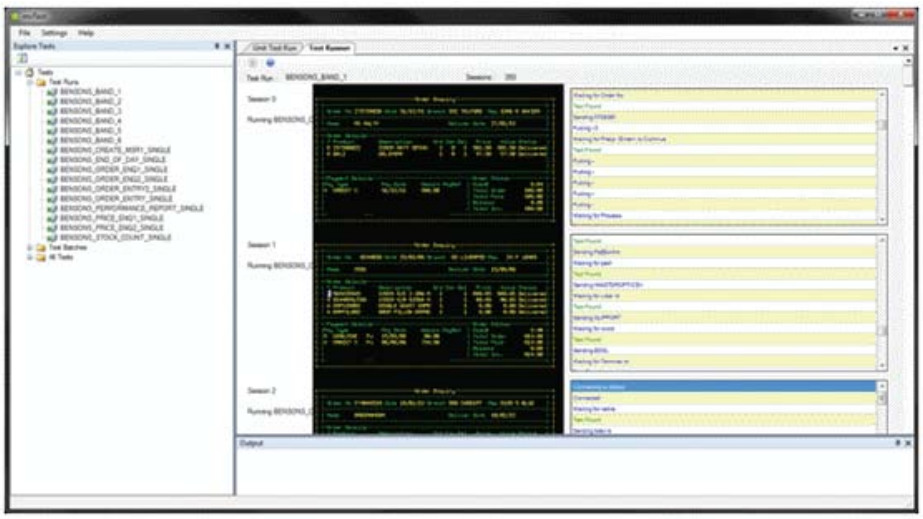
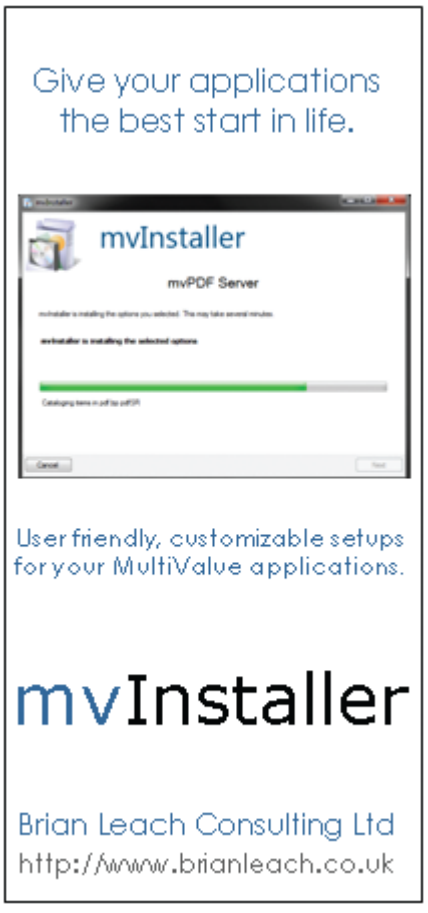


Fig. 2



BY BRYAN SHUMSKY, REVELATION SOFTWARE

Building A Mobile Application With O4W

Part 3

A Quick Review

Over the last two issues, I discussed Revelation Software's O4W web development toolkit, and how it has evolved since its introduction to support mobile devices. O4W Mobile is built upon jQuery Mobile, an open source community supported framework, to provide a device independent user interface that's appropriate for the "mobile computing environment" (with all the differences that entails from the traditional "desktop computing environment").

In Part 1, I demonstrated how you could build a simple display form using the O4W APIs; in Part 2, we took that same routine and enhanced it to provide more interactivity, user input, and advanced interface elements. In our concluding Part 3, we'll show how you can process the entered user data.

Where We Left Off...

The form we ended up with displayed an interactive menu of items we could order at the imaginary Rev Pizza Parlor (fig. 1).

Send In The Data

Our form is now fully interactive, but until we provide a button for the user to click when they want to submit their order, we won't know what they've chosen.

We can use the O4WButton API call to display a button at the bottom of the menu:

```
O4WButton("Submit Order", "BTN_SUBMIT",
      →  O4WMarkedOptions('1'))
```

This creates a button with the label "Submit Order", with the unique ID of "BTN_SUBMIT", and marks it as being the default button (which isn't really needed here, since there's just the one button on the form, but is still good practice) (fig. 2). This button can now be clicked to the user's heart's content…but we *still* won't get their order details until we tell O4W we want to be notified about the button press.

Digital Version On-Line

Scan Me!

The O4WQualifyEvent API call lets us associate events with elements and controls. There are many different events that can be passed in to O4WQualifyEvent, but the one we care about in this instance is the "click" event:

```
O4WQualifyEvent("BTN_SUBMIT", "CLICK")
```

Now, when the user presses the "Submit Order" button, a "click" event will be sent into our Basic program; we'll have to add some code to properly handle this event. In particular, we'll want to read in all the values from the form, save (or process) these in some way, and provide a response to the user so they know their submission was successful.

Obviously, the actual creation of the pizza is beyond the scope of this article, but we can provide an example of what *might* be done in the click handler (fig. 3).

Whenever your routine responds to an event, you must first call O4WResponse() to properly set up the O4W environment. This corresponds to calling O4WForm() when creating the form in the "create" event.

We can then use the O4WGetValue function to retrieve the values that were set in the form. Note that we use the "variable names" of the input elements (and not the "unique IDs") when requesting the variable values.

Finally, we will create a dialog box that provides an acknowledgement to the customer. One of the quirks of O4W (and the underlying jQuery) is that, when we want to respond to an event, we don't usually create new sections and pages; we just change the content of previously-defined sections.

Let's go back to the end of the create event, then, and define a "placeholder" section for our dialog (fig. 4).

We can now replace that "placeholder" section with the real dialog we want to display (fig. 5).

We learned in Part 1 that each "page" of mobile output has sections that each play a well-define role — the "page" role, the "header" role, the "content" role — and when we want to display a dialog, we must create a section that is identified as being in the "dialog" role:

```
dialogOptions = o4wmobileoptions("dialog","b")
```

Fig. 1

Fig. 2

```
65      Case event _eqc "CLICK"
66          O4WResponse()
67          * Load in the submitted order
68          BSTICKS = O4WGetValue("BSTICKS")
69          GBREAD = O4WGetValue("GBREAD")
70          WINGS = O4WGetValue("WINGS")
```

Fig. 3

We pass this style/option value in the O4WSectionStart API call, and then define the rest of the dialog (the header section and the content section). Note that the O4WSectionStart also includes the O4WResponseOptions() option parameter; this tells O4W that this section is replacing a section that was previously defined.

With the dialog page defined, we instruct O4W to display the dialog with the O4WDialog call (fig. 6).

The user has the satisfaction of knowing that some hot, gooey goodness will soon be coming their way, and the O4W developer has the satisfaction of knowing that developing mobile forms is now as easy as…pizza pie. IS

```
55          O4WTableEnd("locTable")
56          O4WSectionEnd("locContent")
57          O4WSectionEnd("locPage")
58
59          * build a section for any dialogs we want to pop up
60          o4wsectionstart("menuDialogPage", o4wmarkedoptions("0"):dialogOptions)
61          O4WText("Thanks for your order!  Your order number is pending...")
62          o4wsectionend("menuDialogPage")
63
64
65      Case event _eqc "CLICK"
66          O4WResponse()
67          * Load in the submitted order
68          BSTICKS = O4WGetValue("BSTICKS")
69          GBREAD = O4WGetValue("GBREAD")
```

Fig. 4

```
71      * Display an acknowledgement
72      o4wsectionstart("menuDialogPage", dialogOptions:o4wresponseoptions())
73      O4WSectionStart("dlgHdr", hdrOptions)
74      O4WHeader("mmmPizza Order", 3)
75      O4WSectionEnd("dlgHdr")
76      O4WBreak()
77      O4WBreak()
78      O4WSectionStart("dlgContent", O4WMarkedOptions("0"):contentOptions)
79      * use standard O4W API to build this output
80      O4WText("Thanks for your order!  We're busy baking it right right...your order number is #":DATE():TIME())
81      O4WSectionEnd("dlgContent")
82      o4wsectionend("menuDialogPage")
83      O4WDialog("menuDialogPage", "Order Received")
```

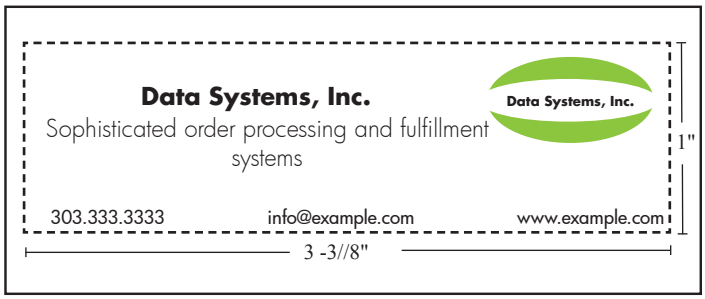
Fig. 5

Fig. 6

MARKETPLACE

ADVERTISER INDEX			
AccuSoft Enterprises	9	Ladybridge Systems, Ltd.	15
Brian Leach Consulting, Ltd.	17	PICK Programmer's Shop	15
Execu-Sys, Ltd.	9	Revelation Software	5
HDWP	19	SJ+ Systems Associates	11

HIGH IMPACT 4-COLOR DISPLAY AD



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 fulfilment systems

- Approximately 60 characters per line

ONLINE BANNER ADS:

VERTICAL BANNER (120X240)

$2,880.00 per year or $300.00 per month

RECTANGLE (180X150)

$2,592.00 per year or $270.00 per month

SQUARE BUTTON (125X125)

$1,680.00 per year or $175.00 per month

E-MAIL NEWSLETTER BANNER:

HALF BANNER (300X90)

$4,200.00 per year or $175.00 per issue

ONLINE MARKETPLACE LIST:

BASIC LISTING – FREE

- Company Page
- Product Page – Limit 1
- ADs will be displayed on your page

PROFESSIONAL LISTING

$350.00 per year/ $35.00 per month

- Company Page
- Product Pages - Unlimited
- No ADs displayed on your page
- Download Links

For more information contact Nathan at: nathan@intl-spectrum.com

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

Clifton Oliver & Associates
www.oliver.com | wco@oliver.com

Drexel Management Service
www.drexelmgt.com | dconboy@drexelmgt.com

Execu-Sys, LTD
www.eslny.com | mh@eslny.com

HDWP
www.HDWP.com | results@HDWP.com

Kore Technologies
www.koretech.com | sales@koretech.com

E-Commerce | Business Intelligence | Integration

Extending the Enterprise Through Integrated Best-in-Class Solutions

kore Technologies

- ☑ B2B / B2C StoreFront & Portal
- ☑ Mobile eCommerce Solutions
- ☑ MultiValue to SQL Data Warehouse
- ☑ Enterprise Reporting & Dashboards
- ☑ Best-in-Class System Integration

Solutions that work. People who care.
866-763-5673 | www.koretech.com | sales@koretech.com

Norcron Inc. (U2/SB+ VAR)
www.norcron.com | sales@norcron.com | 404.459.6500
U2 & SB+ programming experts, Masterpack sales and support

PICK Programmers Shop
www.pickprogram.com | brian@pickprogram.com

Precision Solutions
www.precisonline.com | Kevin@PrecisOnline.com

CREDIT CARD PROCESSING

MV Tech Services Inc.
http://www.mv-tech.net | tom@mv-tech.net | 952.474.3795
Process credit cards directly from any MultiValue application.

DATABASE

Ladybridge Systems Ltd
www.ladybridge.com | sales@Ladybridge.com

Revelation Software
www.revelation.com | info@revelation.com

REPORTING

Brian Leach Consulting, LTD
www.brianleach.co.uk | brian@brianleach.co.uk

Entrinsik
www.entrinsik.com | sales@entrinsik.com

TERMINAL EMULATOR

AccuSoft Enterprises
www.asent.com | pjs@asent.com

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

LETTERS TO THE EDITOR

Have an opinion on an article: Agree, disagree, or enhancement to an article from a previous issue? International Spectrum and our authors are interested in hearing from you!

E-mail: editor@intl-spectrum.com

CLIF NOTES: HELLO, BIG BROTHER

Continued from page 23

ment invasion of privacy to be a bit hypocritical, or at very least, myopic. Everybody seems to be so bent out of shape over Big Brother. But what about its sibling? Big Data?

Big data is one of the latest "new things" in our information services industry. The gist of it is to collect billions and billions of little data points (apologies to Carl Sagan) and then analyze them looking for patterns, trends, habits, etc. in order to predict, target, and manipulate opinions, purchases, votes, you name it. Big data got a big boost in our last presidential election due to its use to target campaigning activities and its successful prediction of the outcome of the election. The extrapolation of this kind of thing to actual voting practice and it's rather chilling conclusion was foreseen by Isaac Asimov in his 1955 short story *Franchise*. (And I will point out that the UNIVAC I system successfully predicted the outcome of an election as early as 1952. So I'm not sure what all the self-congratulatory chest thumping is about.)

And where are these billions of bits of data collection coming from? In many cases, you. That's right. You and your activities generate them. Sometimes they are collected by monitoring and observing you as you go about life. Other times you explicitly give them the ability to monitor and track you. Have you ever signed up for one of those grocery store loyalty "discount" cards that they scan every time you make a purchase? You have just given them permission to monitor and track you. Do you use a Gmail address for your email? Guess what? Yup. Content monitored. Twitter? Data mined. Product warranty cards (an oldie but a goodie)? Did you ever wonder about those questions regarding gender, income level, etc. that have nothing to do with the product in question? And when you make a purchase with your credit or debit card, why does the cashier ask for your ZIP Code? Because it helps them target their advertising, that's why. And isn't advertising an attempt to manipulate?

Don't think for a minute that I am suggesting that we ignore or stop trying to curb government intrusion into our life. I am completely against that. But I think it only fair to point out that government is not the only culprit. In many cases we ourselves are co-conspirators in the demise of our own privacy. We are willing to give up our privacy for the benefit of a few coupons, a "loyalty" markdown on a marked-up product, or to get some service for "free." (TANSTAAFL.)

Big Brother is here, and its name is…

… Walmart. IS

Hello, Big ~~Brother~~ Data

BY CLIFTON OLIVER

Many of you have heard (probably ad nauseum) of Edward Snowden, an employee of a contractor to the National Security Agency who took it up upon himself to divulge a number things about NSA's intelligence gathering activities. The information has caused quite an uproar, not only within the United States, but also in other countries. But Snowden is not the main topic of this column. Some think that he is the highest form of traitor. Others think that he is some sort of hero figure. (Personally, I think he's a narcissistic little pinhead.) Let's talk instead about how people have reacted to the information "leaked."

The word that I think that best describes most people's reactions to finding out that an intelligence agency routinely gathers data about their phone calls, emails, and other Internet activity is "hysteria." Former Senator. and Vice President Al Gore called it "obscenely outrageous." (Really, Al? As the self-labeled "inventor of the Internet," why didn't you foresee this and put in a few protections? I'm just asking.)

Of course, there was the initial attempt to downplay the entire thing by pointing out that only metadata was being collected. In other words, they were gathering information about who called who from where and when, but nothing about what the conversations were actually about. You know, the typical "nothing to see here; move along" technique. But later comments from various officials suggests that they actually do store content but, honest, don't read any of it without a court order. Trust us. Really. (I guess this is what Star Wars fans would refer to as the "these are not the droids you're looking for" defense.)

Although the revelations first concentrated on phone traffic, Eddie later "revealed" that the NSA was also tapping into the Internet backbone and monitoring what passes over that — email, tweets, IM conversations, voice, and so forth. That triggered another round of wailing and gnashing of teeth.

And let's not even get into all of the hypocritical frothing at the mouth and posturing that went on about various countries embassies spying on each other. No duh.

Frankly, I have also been shocked and flabbergasted. But my stunned reaction is not to the revelations regarding the NSA's intelligence collecting activities. I am totally taken aback by the number of people who claim to be surprised by this.

The Government is watching our phone calls and Internet activity? People didn't realize this? This surprises them? Really? I have to admit, I am somewhat aghast and disheartened by the demonstration of the population's naïveté. Monitoring communications is nothing new. It's been going on for years. Google "Echelon."

You never suspected that the Internet is being monitored? Google "Hepting v. AT&T" in which we find

> "It is alleged in the lawsuit that in 2002-2003, AT&T permitted and assisted the NSA to install a NarusInsight system in its San Francisco switching center (Room 641A), which was capable of monitoring billions of bits of Internet traffic a second, including the playback of telephone calls routed on the Internet, and thus in effect spying upon the entirety of the communication of many or all American citizens and businesses who use the Internet." [en.wikipedia.org/wiki/Hepting_v._AT%26T]

And that is old technology, folks. One of the things that apparently the masses don't realize (or at least don't think about) is that any time you use electronic communication or data storage and retrieval, you have in fact given up all expectations of privacy. No, that is not the law. It not right. Monitoring your email without a court order is illegal. But since when has being illegal ever stopped government (or Google) from doing whatever they wanted to do?

But hang on a moment. I find all the frothing at the mouth about govern-

Continues on page 22

The Premier MultiValue User Event

INTERNATIONAL Spectrum®

2014 MultiValue Conference

April 7 – 10, 2014

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

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.

The Wigwam | Litchfield Park, Arizona

www.intl-spectrum.com/conference