

# IBMi Modernization with Valence

Mark Schroeder interviews Richard Milone and Robert Swanson of CNX Corporation



**MARKETING  
ASSISTANCE**  
technologies

## **CNX Corporation**

### **Richard Milone**

Richard Milone has worked as an application developer on the IBM i platform and its various predecessors for 21 years. Currently serving as CNX's Chief Technical Officer, Richard is dedicated to promoting the IBM i profile, encouraging RPG programmers to expand their development skills beyond the traditional green screen paradigm and into building dynamic Web 2.0 applications that run directly on IBM i. Richard has lectured at COMMON, RPG World and iSeries DevCon about the merits of uniting RPG back-end logic with widely available JavaScript libraries on the front-end, a technical "power combo" that produces IBM i-based applications with state-of-the-art browser interfaces.

### **Robert Swanson**

Robert Swanson has worked as an RPG developer since 1992, focused on integrating ERP applications into traditional plant floor environments. With a background in Industrial Engineering, Rob has been focused on weaning manufacturing and distribution operations from "off-line" reports and spreadsheets, moving them instead into real-time, paperless applications that encourage plant workers to interact with office personnel through intuitive software interfaces. Currently serving as CNX's President, Rob is a strong advocate of introducing Web 2.0-style applications into conventional IBM i shops as a means to dramatically increase productivity and bolster the image of IBM i as a world-class application server.

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

Mark Schroeder:

Rob, can you tell me a little bit about your background in application development and software modernization?

Robert Swanson:

Yes, well we have been the IBMi, AS400i series, system I world, what have you, since the early 90s and we were principally in the green screen realm as mostly everyone else in the RPG world was for quite a while. We had reached a point where we recognized that it had pretty much ran its course with the green screen applications and needed to find a way to modernize it. We tried quite a few different approaches and found that none of them really addressed all of our touching points for what we would considered a modern app by today's standards. By modern app I mean something that really looks and acts like a PC application but runs through your web browser. So, that basically is what formed the genesis of Valence after literally years, which Richard can talk to you about. It was literally years of trying different approaches to get the best face possible on our software. We found that we really kind of needed to go on our own and develop a new avenue for delivering the Web 2.0 field that we really felt our product needs and now we have Valence.

Mark Schroeder:

Richard, do you want to add anything more for experiences, company and modernization?

Richard Milone:

I've been on this platform ever since it was released. I guess I'm kind of showing my age a little bit there. Yes, I mean Rob's correct. We've been pretty much trying to find a way to get our green screen applications to the Web ever since the web was

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

invented and never really found a very good way to do it. We were trying every third-party tool, anything IBM had provided and finally we bit the bullet and created our own way and that is Valence.

Mark Schroeder:

What kind of advantage do you think IBMi shops will gain from modernizing their applications? You know, going from the green screen to the Web or whatever approach we think is modern?

Robert Swanson:

I would say, to start with, their user base is going to be more pacified or satisfied with the interfaces, the interface options they have with a modernized application. But generally speaking, the backend is pretty robust and inherently scalable by the running on the I, but limited in the fact that it's running through a text based interface, a 5250 client. So, by modernizing the application you're preserving your backend integrity but giving your users a whole plethora of new options for viewing and interpreting all the data, the wealth of data that the I is so adept at storing, in a way that you never could in the past with the green screen.

Mark Schroeder:

Yeah, I think the Web has been just a wonderful thing for the I actually because like you said, the strength is all it's number crunching. It's just never has had a decent front end.

Richard Milone:

You know when you say what are the benefits of having this kind of interface on your applications, I mean it's almost like, you can't even think that way anymore. It's getting to that point. It just has

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

to be that way. I remember in the early 90s we used to have to do a cost justification on every PC we wanted to purchase. Now today you wouldn't even consider doing that. A PC is just an expense because it's just part of what people need to do their jobs. I mean users demand and require graphical interfaces and are most comfortable with web interfaces because that's what they use at home and you know, their whole lives. It's what people are familiar with. We get a lot of customers who just say it's unacceptable training somebody on the green screen. For example, we have one customer who used to have in order entry application where it would take them basically a month to train a new person before that person could enter an order on their own. Whereas, in a Valence application, you know, they can do that after the first day because the screen is so much more intuitive.... Things like that. It's almost becoming a basic requirement. I may be a little ahead of my time in saying that, new applications anyway I'm talking about. For new applications I really don't know how anyone could justify creating a new green screen application.

Mark Schroeder:

It is true, but you know it is funny. I have seen some companies that are still wanting to go the green screen route which I can't understand.

Richard Milone:

No, they're out there and maybe that just comes down to what their knowledge is in, so that's what they have to continue to go further in the short term. I think we all recognize of course that longer term graphical interfaces and I truly believe Web interfaces are here to stay for a very long time.

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

Robert Swanson:

I think some of those companies might be victims of false starts in the past and have been burned by trying so-called modernization projects that resulted in... it could be a screen scrape solution or a very clunky result that the users ended up abandoning because they found that they were less productive than they were when they were on the green screen. That's probably the number one trap that anyone embarking on a modernization project needs to watch out for. Is that the product they're creating in the application has to be an improvement over the green screen and just because it's graphical doesn't necessarily mean that it's better. There are a lot of other factors that need to be taken into account.

Mark Schroeder:

Since we're talking about modernization, we should define how do you, as a company, define modernization?

Richard Milone:

There are a lot of different definitions of modernization and you could ask this question to a dozen different quote unquote experts on the I and you're going to get a dozen different answers. I like to think of a modern application as very generically, one that the user can walk up to and figure out how to use it without any training whatsoever. You know, that's a very generic modernization definition, but it's the one I always start with and one that we keep at the top of our minds when we're designing applications. In other words, you could take a tool like to Valence or another web development tool and you could create an application with that...that looks nice but is totally nonfunctional, non-intuitive. So, there's a big design element to it. Just because the interface looks pretty does not mean it's necessarily defined as a modern application. There's also other interesting definition

Copyright © 2010 by Marketing Assistance Technologies LLC

*No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.*

of modern application. One might say a modern application is one that is modular and easy to kind of interface with you know.... I might have a bunch of backend, different service programs or whatever, that I can hook up to any kind of front end interface and I may define those as modern because they have a modern quote, unquote object oriented design and so forth.

Mark Schroeder:

So it's not just using a modern language to modernize.. it's the whole approach?

Richard Milone:

That's a dangerous path to get down. To talk about a modern quote, unquote language. You can look at RPG and there's so many different flavors of RPG that you can't just say something like RPG is old or RPG is modern because it depends on how you use it. If you're talking about an RPG2 program, it's all columnar, that's heinous. You're talking about a free format RPG program that is free format, fully proceduralized, using activation groups and service programs... that's pretty darn modern. You know, I could put that up against any other language.

Mark Schroeder:

Right, RPG is quite a good language, especially with the upgrades they've made to it. You had an application that you guys needed modernized. What were the reasons that you were looking at modernizing your application?

Richard Milone:

Well, I mean, the first reason we wanted to modernize was because we wanted to get more sales. You know, it's very difficult

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

these days to actually sell a green screen application because the people that you're selling it to.... You know, you go into a room of executives and you want them to pay money for your software it has to look modern. It's hard to even get the green screen apps in the door these days, so you know, that was the biggest reason.... And of course, you know, what we talked about earlier in terms of the user experience and everything. I mean, we want the users that are using the application to be happy. Of course number one, our applications are trying to solve business problems... But number two, we can solve those business problems easier if the users are happy using the applications. It's difficult to get users happy using a green screen.

Robert Swanson:

Yeah, and to add to that, a byproduct of modernization we found unexpectedly and surprisingly, is that it's actually faster, in many cases to design apps that run in a modern interface than to do the green screen equivalent. For example the Web 2.0 equivalent of a sub-file is called a grid and we can program one of those in minutes without having to worry about indicators, paging rules or folding and all the other things you have to think about when you're doing design for a green screen. Not to mention the real estate limit that you have in the green screen realm of 80 columns....or 127 if you have a wide mode. Whereas in HTML all that stuff is kind of automatic you know. It will automatically spam the width of your screen and beyond as needed and we can list as many details as we want. You can even give added functionality to the users and let them move columns around or turn them off, sort by them, and all these that you really couldn't even fathom in the green screen world. The beauty of it all is that we can actually do a lot of this coding and design in a lot less time than it took to do a much less impressive green screen program.

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.



Richard Milone:

Let me qualify what Rob said there because many of your listeners, you know, will be quite skeptical of what he just said there. It really is true...you have a learning curve of course... but the key thing that I think he might have left out is that it really depends on using the Web 2.0 type of design methodology that Valence brings. You know, either using Valence with its design methodology or something similar because what most people may be familiar with is the older Web 1.0 design methodology where you're trying to generate HTML pages within your RPG. That can be quite difficult to maintain the state between pages and so forth... A lot of problems with that and it creates a lot of new work that you didn't have to deal with before when you were doing green screen apps. With the Web 2.0 design methodology that Valence brings, this state is maintained on the front end and all of these issues about maintaining the state between pages and all kind of evaporate. All of what Rob was saying about, you know, giving the drag-and-drop functionality and sorting columns and moving things around. That kind of thing is mostly automatic so that the developer can really bring the rich functionality to the user with a lot less time. It's going to take time to move people to this design methodology and to gain experience with it.... But after six months or maybe a year in some cases, you will be designing these types of applications faster than you could have on the green screen.... And of course adding a lot of functionality wouldn't even be possible on the green screen.

Mark Schroeder:

That's the really nice thing about designing, building with Web 2.0 and having a tool that will provide you with that ability to add that functionality. I hate the idea about having to go in and code

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

everything. Having a tool like that makes a difference. Now, you looked at some different options before you started your modernization process. What are some of the different modernization options you looked at?

Richard Milone:

Well, I guess I'll have to take this one because I am the Chief Technology Officer at CNX and it's kind of my job to evaluate technology options. You know, I don't really want to name third-party packages, but we try, let's just say, we tried pretty much every one of them out there extensively. I guess I could name some of the IBM ones. I mean, I was trained in Java right up in Rochester and developing in Web sphere. In fact we deployed Web sphere applications for certain customers in the late 90's and early 2000's... in that timeframe... but never really kind of felt comfortable with it. We could do it, it just seemed like it took a long time to develop applications with it and to make any kind of change whatsoever seemed like a very long, drawn out, deployment process. So, I mean the bottom line is, we just never really felt comfortable doing it and we could kind of see that it wasn't going to be a long term option for us. So, eventually we dropped the Web sphere and the Java and the Java route. When we did that, that's when we started kind of evaluating the third party tools and other ways to do things, with a particular interest in seeing if we could continue using RPG, because we really kind of felt like we had a lot of RPG knowledge in that it worked well and it was stable... and the only real problem was the interface. You know, in fact if I could expand on the answer a little bit... I can tell you that in early 2005 was when I had the epiphany. One of my customers told me to check out Google maps and I had been a longtime user of the Yahoo maps, which at the time you kind of had to page left and right and wait for the whole page to

Copyright © 2010 by Marketing Assistance Technologies LLC

*No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.*

regenerate. The first time I used Google maps, again early in 2005, it really shocked me. I went to the webpage and I quickly found that you could slide the map around and I was like... wait a minute, I didn't really install any application on my computer and now my browser is working like a PC application and I really got interested in how they were able to achieve that. It was Google map that was really one of the first applications that came out that used a kind of full Web 2.0 design methodology, with Ajax and all of that... and a lot of rich JavaScript in the interface. So, I kind of said about dissecting how that works and that was really kind of the light bulb, if you will, that kind of set us about creating Valence.

Mark Schroeder:

You also looked at some web facing options, is that correct? What kind of issues did you see with Web facing verses what you've gone to?

Robert Swanson:

As I recall, the Web facing options that we tried were very clunky. Performance was less than we deemed acceptable and we were always operating on the knowledge that the underpinnings of our application where all rooted in the 5250 display and no matter what kind of grand ideas you have for the front end, if you're writing it over 5250 interface, you're constraining yourself. We really didn't want to shackle our application with that kind of hindering, going forward and that's why we didn't stick around with the Web facing solution for a very long. We did give it a good Boy Scout try though.

Copyright © 2010 by Marketing Assistance Technologies LLC

*No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.*

Richard Milone:

The last time we were serious with Web facing or seriously trying it was, say like 2000 to 2003 timeframe...maybe into 2004. There are much better tools available today, but like Rob was saying, you are still locked into the 5250 DataStream. A tool like valence brings you into full Web 2.0 design methodologies which is event driven. I mean, you just simply cannot develop a completely world-class, rich Internet application with a screen scraper or a Web facing type tool. Basically what you wind up with those kind of tools is something that looks a little bit better and works somewhat slower. You know, that's what you wind up with. You know, you do not change the underpinnings. Developers that get themselves involved in really long-term projects of Web facing type projects really don't add to their value. I'm talking about their personal value in the IT marketplace. You can put down on your resume that you learned how to use a Web facing tool. That means almost nothing to a prospective employer. If you put down your resume that you developed applications that used full Web 2.0 design methodology.... You know, that is worth a lot on your resume. It doesn't even matter that you're using RPG as the backend. Now we're talking about learn the front end stuff and this whole design methodology and it's worth a lot in the IT marketplace.

Robert Swanson:

It makes it kind of platform independent.

Mark Schroeder:

... and as developers that's something we're always looking for, is we want to increase our own personal worth in the marketplace as well. We may not ever move to another company.

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

Richard Milone:

Yeah I mean I'm not a sales guy or anything but I talk to developers that are interested in finding something. I really try and drive that home. I mean, when you look at something like Valence, it brings the design methodology that has been very successful in other platforms... especially with back ends of Java in PHP and other languages. You know, we bring that to the eye with RPG on the back end. That's essentially what Valence does with a lot of tools.

Mark Schroeder:

That should make it a pretty good move for someone that does know RPG....to use a tool like yours and move to the Web.

Richard Milone:

An experienced RPG programmer using Valence is going to have a very short learning curve on the backend. We draw a nice distinction between backend and front end. On the back end stuff, as long as they know how to use a service program and can read free format RPG.. Or maybe that's something they want to learn and you know, Valence could be a conduit for that...because there are a lots of examples in that. There's going to be a shorter learning curve on that... and on the front end we tell people straight up, you know you've got to learn the Valence front end. What we do, is we provide all kinds of examples and all kinds of training.... Mostly for free... to help RPG programmers learn that stuff. All of our training and all of our documentation is all geared toward the RPG programmer's perspective.

Robert Swanson:

As an interesting aside to that, I'd like to add that we have found that hiring out of college today... you know, students or graduates

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

that are fluent in the concepts of Ajax, Web 2.0 and JavaScript are very quick studies to RPG. RPG is a very structured language that someone that's learn some of the more complicated means to achieving what RPG does, will find refreshing. Once they get over some of the quirks of RPG, such as, aligning your D specs in the right spots and things like that, they are usually off to the races quite quickly. We've had some success with getting some people that have never heard of the I, up and running.... Doing RPG back ends and Web 2.0 front ends in a matter of weeks surprisingly.

Richard Milone:

Yeah, that is a very good point Rob. You know, we have experience in going both directions. We can take RPG programmers who are already very good with the backend and train them on the front end stuff... and you can take people that are experienced on the front end stuff and train them on the backend stuff. We also have very good experience in keeping those roles separate. We have people here within CNX, in our services group, who do nothing but the backend... and we have people that do nothing but the front end..... And the two are interfaced together. I always liked knowing everything. I guess I am just that kind of person. I have to know everything, but inside your organization you can maintain those functions separately and it does work out. You might think at first glance, well you might not wind up with the great application if you keep the front end people in a separate section in the backend people separate. You know, they interface together and come out with a common, handshaking protocol and they can design applications well that way.

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

Richard Milone:

Which all goes back to the whole paradigm of keeping your business logic separate from your display logic. They really can operate as independent teams quite effectively. You can hammer out new applications quite quickly when everyone understands their role as a team.

Mark Schroeder:

That such a good concept to follow because then you can have the two different teams working, pretty much in sequence with each other.

Richard Milone:

Right, it gives you some parallel development.

Mark Schroeder:

Exactly.

Robert Swanson:

Right, I mean we have, many times with new customers of Valence, they will contract us to write a portion of their applications. That's part of how we make our revenue with Valence. We have a good service, this team here and if they have very experienced RPG programmers and they understand the Valence stuff right away on the backend.... They do that portion and they can contract with us, at least for their first application as their guys are still learning. We can normally do the front end parts with them and that works quite well. We do the front end portion, we work with them to do the backend portion. At the same time, they are learning the front end.

Copyright © 2010 by Marketing Assistance Technologies LLC

*No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.*

Mark Schroeder:

When you decided to build Valence, what are some of the capabilities you felt were important to include?

Richard Milone:22:24

The JSON formatting capabilities are really kind of the first thing. You know, any kind of tool that you're going to have that has a full-blown JavaScript front end.... Of course Valence, the front end is based on EXTJS. That JavaScript framework. It's native language for communicating data is JavaScript Object Notation. So the RPG core of Valence had to have very good JSON formatting capabilities. So that was really kind of the first. It had to do that automatically and very easily for the developer, otherwise you spend a lot of time developing JSON formatted strings and not enough time in your business logic. Beyond that, the portal, the Valence framework comes with a full-blown portal. Which we call the Valence portal. And that handles the login. So you know, when your users need to login, when you have to authenticate or authorize your users to certain functions and programs that you've created in Valence. You know, this is normally the first thing that a web developer gets confronted with when they want to put up a Web application and a lot of tools when they are trying to do this on the I... How do I get the user to log on because we don't have a green screen?.. We don't have a quote unquote sign on screen. Valence has all built in tools for that. That's an absolute requirement. Login control, session control and the ability to authorize users to certain functions. A menuing system if you will. You know, I don't want to dumb down what the Valence portal is but it's handling all the login, authentication and menuing.



Mark Schroeder:  
How do you handle security in Valence?

Richard Milone:  
Well, the browser will really only communicate in Ajax with the domain that it originally called for. There are ways around that. So Valence doesn't really open any security holes in your browser that don't already exist on every other platform or tool. For the Valence portal, the authentication in order to get logged on, is pretty strong. You know, there are two login mechanisms to the Valence portal that you can choose. You can choose to have Valence specific user profiles. So Valence would maintain its own separate database of users aside from your system I profiles... that you can then cross reference to your IBMi profiles. In other words, you can have like 100 external users and then you could cross-link them to one specific IBMi user if you didn't want each one of those that have an IBMi user. Or you could turn on the IBMi user ID authentication in the Valence portal so you had to have a specific IBMi user ID. Once your user ID is authenticated, you get a very strong session key sent to your system and on the backend that session ID is stored along with a whole bunch of information about who you are, what IP address you came from and all that. Then every single subsequent call to a Valence program has to have that session ID authentication on it. Everything is authenticated on every single request. We've made a very, very good effort. I mean, it sounds like at first glance that could be very slow. You've got to re-authenticate on every single call. It is the only way to keep the browser secure or to keep the session secure because it's a stateless connection, you have to authenticate on every call. We've gone to great length to make it efficient and fast.

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

Mark Schroeder:

Yes, I agree with you because it is stateless, you have to authenticate every time and that really is the only way to make something secure there.

Robert Swanson:

Yes, the key is how do you know, as Richard said, the means by which you go through that authentication can either be taxing or we can make it as efficient as we can which is what we really focused a lot of resources on optimizing that process.

Mark Schroeder:

It sounds like the developer doesn't even really have to think about that. That Valence actually takes care of that for you?

Richard Milone:

That is automatic. If you think about, you know, a call from the front end to an RPG program on the backend, it's just like calling any program. You're calling a specific program name and you're sending it certain parameters. Well, the Valence portal automatically inserts the session ID parameter on every call so that the developer doesn't have to add that. It inserts a bunch of other stuff too.. So, that kind of authentication and handling that, is kind of transparent to the developer... it's automatic.

Mark Schroeder:

That's good, when I started doing web developments, handling sessions and so forth... that became one of the bigger issues because you didn't have to do that in the green screen or other type of programming. It's like Web developing was a backwards move in the whole development paradigm.

Copyright © 2010 by Marketing Assistance Technologies LLC

*No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.*

Robert Swanson:

That's why I think a lot of the RPG programmers, when they get their hands on the Valence, it's kind of like a breath of fresh air because it's actually something that is designed from their perspective... because, in fact, Valence was designed from the ground up, by a group of RPG developers.

Mark Schroeder:

Now, what about the Ajax code? I know that Valence includes a lot of it with the tool. Are you allowed, or are you able to write some of your own Ajax code and does it save it or does it overwrite your code? How does that work...or is there even really a need that you find to write your own?

Richard Milone:

Well, I mean when you say Ajax code, that can mean a lot of things. I mean essentially, in the Valence framework there's an Ajax request program or function within the front end JavaScript that you can call...that basically lets you call any RPG program and send it parameters. Then you get back a response and you do something with that response. It's pretty simple. A lot of times the wording of the Ajax scares people. It's like they've seen it in the past or seen video demos or how to do it...

Robert Swanson:

They think of the Mr. Yuk stickers... when they were growing up on the bottle of Ajax.

Richard Milone:

Right, I mean you know, the way it's done in valence is pretty simple. I mean, you're effectively just calling a function that calls in RPG program and send parameters, weeds for a response and

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

then does something with that response. It's pretty simple and yes, there are... when you talk about... when you ask the question about what you can do? Can you change what we've done in Valence? Valence is a framework and the framework includes the RPG toolkit on the backend, a JavaScript framework on the front end and it includes a Valence portal...which in effect, the Valence portal is really a Valence program, okay? So without the Valence portal the front end and back end stuff would still work to create Valence programs. The Valence portal really takes a lot of the work of what we were just talking about. Signing on, session management, automatic handling of certain things. The key is, it's really hard to describe how you would create a program in valence other than to just look at the valence examples. Valence delivers dozens of different examples that RPG programmers can look at and the code is described very well. So a lot of us like to code by example. So, if you like to code by example, valence is a great thing to try because you know, there's lots of nice examples in there. A lot of the examples mimic what you might want to do or what you might have done in your older programs. For example, everybody knows about sub-files and everybody has had to deal with them at some point. Some people do load all sub-files. Some people have gotten into a page at a time sub-files. There's direct analogues to those in Valence already done in certain examples. So, it's a pretty simple thing to take those pages of time to sub-files and in Valence and they are called paging grids... and to just take those, you know, and copy them and make them work over your own files. So it's easy to be successful with Valence pretty quickly, just by looking at the examples.

Mark Schroeder:

What are some of the basic steps that are used in a modernization process like from start to finish... to put a screen up?

Robert Swanson:

Let me start and I'll let Richard add to that. I think that you have to be careful with that approach. That you're not just taking an existing green screen application and thinking okay now I want this to work in a browser. I think you actually have to take a step back and think about what is the objective of this application? What is the user looking to get? What is the system looking to store from the user?...and kind of start that way because the whole Web 2.0 world gives you such a dramatically different approach to programming. It gives you so much more flexibility and freedom in your design. You no longer need to be constrained by the procedural page, the page to page or window to window that you are necessarily working in on a previous RPG green screen example. So I would say the first thing to do is just basically list out what is the objective of the program and what are the things that the users are looking for? How often are the users needing to leave your application?.. maybe on another session or drill down the function keys or so forth to get the information they are trying to get out of this system and how can you convey that or present it in a way that's more user-friendly and just right there and intuitive... and then take it from there.

Richard Milone:

Yeah, I mean I'll add to that. You know, one of the typical scenarios we deal with here at CNX, because we do a lot of services work. We write.. Not only do we create the Valence framework but we also do write a lot of applications for our

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

customers. So, very typically, especially with new customers.. They will give us requirements for new applications and their requirements will look exactly like their existing green screen application. So, you know, we kind of have to hold their hand a little bit and say well, you know, you're not really going to benefit that much by just taking your green screen application and designing something that looks exactly the same way and flows exactly the same way in Valence because you have to kind of breakout of that green screen mindset... because you have things in Valence like tab panels. Where before you may have hit a function key or some other functions or take an option next to a record. Now, you know, you can right-click on something and get a context sensitive menu to do various things... or you can have a tab panel and just have the user click on a different tab panel to look at certain things. You can use drag and drop, which is totally inconceivable in a green screen. So, like what Rob was saying, we really like to convince customers that they should re-engineer the entire application. I mean, if you want to just take the green screen application and move it to the Web, you can use a web facing tool... but if you really want it to be modern and have a modern design where the front end is separated from the backend and uses an event driven methodology... you know, I don't want to get too complex and this interview because sometimes it scares people... but I really feel like I need to mention that, in the Web 2.0 design methodology where the front end is really in control..., you can have. You know it's event driven and more multiple events can be happening at the same time. For example, the user can click on a graph in one area of the screen and that graph could be refreshing and at the same time they could click a drop-down box on the screen in another spot and that could be prompting a combo-box to be refreshing. At the same time, a created in another area of the screen could be refreshing. You

Copyright © 2010 by Marketing Assistance Technologies LLC

*No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.*

know, it's kind of difficult for RPG programmers to think this way at first, but in Valence, in effect, you can have more than one area of the program running at any given time because it's all event driven. When the user takes an action, a section of code runs against that action. That type of design or that kind of thing is impossible to achieve unless you re-engineer your application.

Robert Swanson:

Right, and it really serves to emphasize the importance of separating the business logic from the display logic. So typically you'll take... you might have a monolithic program with some subroutines or procedures in there that you can use, but I found, surprisingly, a lot of the code in most green screen applications is user interface and intertwined and it's... you wind up not needing a whole lot of your code because, depending on how complicated your trying to work this sub-file technology and so forth... you may find that there's just a ton of... a big chunk of your code is no longer needed because it's all based on trying to manipulate a green screen to do what you needed it to do. All that becomes basically unnecessary when you're looking at a Web 2.0 design. You can put all that work on the front end and just focus on the business logic on the backend... which is really what RPG does best.

Richard Milone:

Yeah, I mean RPG is really good at getting data to and from the DV2 database and also crunching business logic. Bottom line is, typical scenario... let me give you an example. Order entry application that we wrote for a customer. They had an existing green screen application. We don't do things like take that existing application and try to modify it to work with Valence. We create a new application...it would be a new RPG program. What

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

we do, is we look at the old program and say, what functions can we re-use out of here? In this particular example, what we did for a customer, we were able to preserve tax calculations... there were a lot of subroutines.. Subroutines in there that calculated things like tax or whether an order to go on credit hold or not. We brought those into procedure within the new RPG program that was based on Valence. So, it's not like you're starting from ground zero. If you have an existing RPG program you can reuse pieces of it and when you do this process.. That's really modernizing your RPG application... because you're creating a new RPG application, you are reusing the business logic from the old one and you're getting rid of that green screen specific code in the old one, at the same time. So what you're left with is a really modern backend, RPG program hooked up to a really modern front end, where the front end and backend are separate to the point where you could theoretically, in the future, attach some other front end to the backend code because there's nothing inherently browser specific I suppose in the code. You know what I'm saying? Any front end could be hooked up to it.

Mark Schroeder:

...or any backend could be hooked up to your front end.

Robert Swanson:

Yes, and you can look at it that way too. Valence is specifically geared for an RPG backend...but theoretically yes, you could hook the front end up to any kind of backend as long as it was giving the same responses



Mark Schroeder:

Have you included SQL along with RPG as a way to access a database?

Robert Swanson:

Oh yeah, SQL is heavily utilized in a Valence. We use SQL very extensively for loading combo boxes and for loading grids of data.. Which would be equivalent to a sub-file. In fact, loading grids of data is extremely easy with Valence... also, it gives users the ability to kind of click on the top of any column and have that column sort. That's all done with the magic of SQL. When the user clicks on a column and that sorts...most developers look at that and say "wow that is really cool."... but it's really not that hard because all it's doing.. When you click on the column it just sends a new parameter to the backend and when the SQL statement is constructed... it just uses that column name as the sort parameter in the SQL statement. It's really pretty easy to execute. Now, you can normally do things more than one way in Valence. You can use SQL and you could use Native IO. It just depends.. There's functions in the Valence RPG kit for both. Typically we always start out with SQL because it's so much easier to program. In Valence you can literally just call a Valence procedure with an SQL statement and return the response back to the front end and have it displayed. In cases where we find that in performance testing the application, if there is an SQL performance issue... For example if you want to give the user the ability to click on any column in a grid... that can have a huge performance impact... negative performance impact, especially if you don't have indexes over those... the correct fields. So, you know, then you can always move to Native I/O if you need to. Normally we start with SQL and see how it performs in testing. Then we can move to Native I/O if we need to.

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

Mark Schroeder:

Nowadays, SQL is really performing quite well on the IBMi.

Richard Milone:

Yes, big, big improvements. I'm talking about like sometimes we will run into customers that have just millions and millions of records on file. Sometimes when you're just getting a handful of records from SQL... it doesn't perform too well. That's where the Native I/O can sometimes get it way faster.

Mark Schroeder:

Right, there are some times for that.. And that's good that you have the option to go either way.

Richard Milone:

Yes, but what's really nice is... and I'd like to reiterate this again... is that we always use SQL to start because many times we have to prototype something to show a customer. So, you know, the SQL is easy, as where a Native I/O... you know, you've got to do a little bit more work on that. You can prototype something with the SQL and even though it doesn't perform exactly the best you can still have it show the customer what the functionality would be. Then if they say, "yeah, it looks good...go forward"...then you can put the time in and switch it to Native I/O.

Mark Schroeder:

That's true. It gives you a nice prototyping tool that way. Another thing that applications need to do a lot of times these days is integrate with other applications. So, say you have a Web application that is not Valence.... I'm assuming you can also

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

integrate and work with those other applications.

Richard Milone:

Well yeah, right out of the box you can use the Valence portal to integrate any other webpages that you might have done in some other third-party tool using CGI 2. We have a number of customers that are doing that. I mean right out of the box the Valence portal is a fully functional portal.

Mark Schroeder:

Is it written to the portal standards so that any portlet can be plugged into it?

Richard Milone:

Well, no... portlets are in Java..so no, you can't plug a Java portlet into it. A Valence portal is essentially a tab based portal that launches a frame within itself to run any webpage that you can call on another link.

Mark Schroeder:

Okay

Richard Milone:

So if you've already developed CGI 2 programs that you can call on any path...or, you know, some other third party tools. As long as you can link to it on the Web, you can kind of bring it under the control of the Valence portal and have it run within the frame of the Valence portal . It is very easy to understand. Most...99% of developers have no idea what a Java portlet is and what the Java portlet standards are...but within five minutes they can figure out how to link in existing pages under the Valence portal. So, right out of the box, they can use that functionality.

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

Mark Schroeder:

So, if someone's going to get involved with Valence... is it just one product that they buy that has the whole package of what they need to use in order to develop? Or do they, are there different options of what they can purchase?

Robert Swanson:

Yes, everything they need is in one download... and you say buy...but technically they can actually use Valence for free. We've added a new community developer license which basically allows anybody with an I to download and use Valence free of charge for as long as they want, provided they agree to the terms of the license which states... if they are to deploy any applications for live use by customers or users, then they would either post the code behind that application onto our public forum as a pro-quo exchange of code. Or they can simply license software at a very reasonable rate, per developer or a different basis that is specific to their organizations.

Mark Schroeder:

What would be the best way... aside from getting the free tools here to use... for someone to start learning how to program in Valence?

Richard Milone:

Well, it's very easy to obtain the tool.. You can just go to [cnxcorp.com](http://cnxcorp.com) and go to the Valence store. There's a free download there, which is the community developer license that Rob just mentioned... that's the free one... and it's extremely easy to install. You download one executable file to your PC.... You launch that executable.. it sets up a remote connection with

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

your I and loads everything you need on there. The Valence portal, the RPG toolkit the front end stuff... and then you know, you have instructions. I mean literally, within 20 minutes of downloading it you're up and running. It's installed, up and running and once you have it installed, you have everything that you need there. There's a Valence manual that's pretty long. I think it's like 125 pages of really great information there. If you're going to learn Valence... you really would want it loaded into your system because you're going to have access to the dozens of examples that are included. You'll be able to look at the code for those examples that you've loaded and you will be able to change things around and see the impact and really explore it once you have it loaded on your system.

Robert Swanson:

I think that's one thing that you will find very intuitive... how easy it is to pull up the code and make changes. I think that one of the things that Web 2.0 really offers as a benefit is that you can make all sorts of changes to the front and not have to compile anything. The effects are immediate, so you can't really kind of learn on a trial and error basis... you know, just keep tinkering with the code and things happen. People are typically amazed at how much effect they can have on the end application, with minimal adjustments on the code that comes off of the example programs.

Richard Milone:

We get a lot of e-mails actually, that say "I've been looking at Valence on your website, I've been reading some things about it... where do I start? I always point people right to the download. We don't even ask you for any information. You can go on there and you can download the software. The full source code is included. I suppose it's a bit on the honor system.... We were

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

saying that the license, if I could reiterate what Rob said,.. You can use the software as long as you want and explore it, learn it and then there's no charge whatsoever unless you #1 ..want support. We do have support packs that are available on our Valence store ...our website at [cnxcorp.com](http://cnxcorp.com) has a store where you can buy support and you can upgrade to the professional developer license... which essentially just gives you the right to deploy your applications to your end-users and keep your code private. The licenses are very inexpensive... it's a per developer license, it's \$595 per developer... and if you license like 5 or 25, you know, there's different discount levels. All of our licensing is done online. You can go right to our website and purchase support or a higher level license if you need it.

Mark Schroeder:

Do you provide training packages also?

Richard Milone:

Yes, we do training.. We do a lot of our training one on one. Very few customers want to actually come to our office for formal training. Normally, when that does happen it normally devolves into a development session anyway because customers normally, when they're looking for training...it's because they have an application that they want to create with Valence in mind. So all of the questions and what they want to gear their training towards, tends to evolve around that particular application that they have in their minds at the time of training.. So what I typically recommend to customers is lets say... let's not even do a formal training session. You know you can buy time from CNX and we will help you create your application. That seems to me, to be the best training mechanism... you know over my many years of experience with this... is we just hook up with customers and we

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

provide one of our developers or more on a team of developers that includes those from the customer... and you know, that you learn more by going. We do formal training sessions, both at the CNX office and at customer sites if that's what the customer wants. But like I said, it's normally more efficient to just kind of get started with an application and purchased some expert's services this time from CNX for assistance.

Mark Schroeder:

Could you go ahead and give me the best way to get a hold of you guys if someone wants to either get some service or buy the professional product or just get more information about everything from you?

Richard Milone:

You can do everything entirely off the website. We have gone to great lengths to make it a very hands-off experience for us in order to keep our costs down.... And they can fill out the form and order support packs, expert services, licenses, everything... even using a credit card if they'd like.

Robert Swanson:

Yes, that's exactly right. I mean cnxcorp.com is going to have a tremendous amount of more information than what we talked about here. So, if people are hungry for more they can go there. There are actually three hours of videos on our website, in terms of total runtime. They are different videos, but added all together, there's about three hours absolutely free right off of our website. Most of those videos are just kind of me going through sample programs and describing them and bringing people through Valence, so people can seek that out on our website quite easily. Like I was saying you can download the framework rate from our

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

website and install it in your system and be up and running within 20 minutes. If people would like to contact CNX directly they can send an e-mail to [sales@cnxcorp.com](mailto:sales@cnxcorp.com) Rob or I usually pick those up with a bunch of other people here.. So, it might not be the best thing for people to try to contact Rob or I together directly because a lot of times we're out or at customers site's or conferences. The best way would be [sales@cnxcorp.com](mailto:sales@cnxcorp.com) ...and that would be the fastest way to get an answer... or alternatively... I would like to recommend this as well, we have a really, really nice forum. When you go to [cnxcorp.com](http://cnxcorp.com), there's a big link for our forum's ,which are growing in number of users and number of posts very rapidly. Especially since the release of Valence 2.1 on March 1. There have been a lot of new sign-ups.... It's totally free to register, to post questions on that forum and other people from the Valence community can answer. CNX development team monitors the forum closely too and always puts up some good answers there as well. So that's another good resource.

Mark Schroeder:

Sounds like you have plenty of good ways for people to contact you or start using the product right off the bat.

Robert Swanson:

Yes, that's exactly what I would encourage them to do too... use it right off the bat. Get their hands dirty, jump into the code and just immersed themselves in the Web 2.0 methodology that Valence backs and I think they will find that in no time.....we just had a customer app demo put on our site last night and one of the questions on it, Richard did the interview with a developer and one of the questions that was answered in the video is how long did it take you to develop in this. I must say the app that this

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.



customer developed with in Belgium is one of the most robust apps I've seen. He went from nothing but RPG background to being, what I would say, a top-tier, Valence developer in the span of... what was it Richard? Four months?

Richard Milone:

Yes, four months. The application that he developed was essentially just himself and half of the time of another developer. So, when you look at this application and you say to yourself this application was created in four months with one and a half developers, you really are stunned.

Mark Schroeder:

Did you say you have a link for that online so people could look at it?

Richard Milone:

Yes, it's in the customer video demos area of our website. It's very easy to find. Just go to the main [cnxcorp.com](http://cnxcorp.com) site and click on Valence there's a another link for the customer video demos.

Robert Swanson:

Yeah, we'll be adding some more over coming weeks too.

Richard Milone:

Yeah, we put out a call in one of our newsletters for customers to contact us if they would like to quote unquote "show off their applications." So, were starting to get feedback from customers. A lot of customers are out there are doing very interesting applications that we've never heard from... so were getting feedback from customers now and will be putting more of those videos as time goes on... as customers contact us to show off

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

their stuff.

Mark Schroeder:

I'll add some links on the [softwaremodernization.com](http://softwaremodernization.com) so they can find your location also..

Richard Milone:

Excellent

Mark Schroeder:

Well those are the basic questions that I had for tonight. Do either of you have anything that you want to add before we close?

Richard Milone:

I would like end with one thought.. Anybody listening to this really should try it out. I mean, there's no risk, it's totally free to try and you can use it as long as you want to learn. Like I was saying earlier, learning something like this can really add to your value in the IT marketplace. Scott Clement has been saying for the last year on a lot of his posts and at conferences and things that RPG programmers should really get out there and try something... the problem is that everybody is sitting around waiting for the best thing to come along... just try something.. I hear a lot of people ask me questions at conferences and by e-mail and things like that. They claim to me that they will never be able to learn how to do what were talking about and that really shocks me because at some point in the past they learned RPG initially. I would contend that this stuff is no harder to learn than when you originally had to learn RPG from scratch. So, you really just have to have the will and the drive to learn it. There are plenty of people out there, all around the world, that have learned Valence in the last couple years. All by themselves, with no help from CNX. And all they had

Copyright © 2010 by Marketing Assistance Technologies LLC

No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.

was an RPG background. So, it's definitely possible and all the resources are there for you to learn it and it is definitely possible.

Robert Swanson:

Yes, one of the more rewarding things I think a lot of Valence developers get from their users is the somewhat incredulous response of their users when they're told that their new application... that they are playing with in Valence.... Is actually running on their old AS/400.

Mark Schroeder:

Thank you Rob and Richard for your time. I hope that you get some people coming to try out Valence after this.

**Copyright © 2010 by Marketing Assistance Technologies LLC**

*No part of this publication may be reproduced or transmitted in any form or by any means mechanical or electronic, including photocopying and recording, or by any information storage and retrieval system, without permission in writing from the Publisher. Requests for permission or further information should be addressed to the Publishers.*