

IBM Podcast

[MUSIC]

MATHENY: Welcome to this IBM podcast. What's happening with Jazz? Today's podcast is a conversation with John Wiegand and Kartik Kanakasabesan on Jazz strategy and results. I'm Angelique Matheny with IBM.

In the words of Sir Winston Churchill, how ever beautiful the strategy, you should occasionally look at the results. In this podcast, Rational Chief Architect John Wiegand, and Jazz Jumpstart Manager Kartik Kanakasabesan, explore the most frequently asked questions about Jazz strategy, architecture and results in order to help customers chart a clear pathway to success.

Hi, John and Kartik. Welcome to the podcast. Thanks for joining us.

WIEGAND: Hello, Angelique.

KANAKASABESAN: Hello, Angelique, thank you.

MATHENY: It's been two years since Rational introduced the Jazz platform. How has your thinking evolved about the Jazz strategy and deliverables?

WIEGAND: Great question, Angelique. Our vision has held the same: the vision of a seamless experience across the application lifecycle, making the...our customer experience much better than it's ever been before [visioned] over these two years from a glimmer and a hope, focusing in on the area of development characteristics, moving on to much broader now, where we support requirements, we [do] our quality management, and much more, much more coming.

I step back and say, what are we learning, and what are...what are the things that we didn't see prior? I would draw attention to three characteristics, I think. The first is an even greater focus on integration, realizing that our customers have a wide range of tools that they're using, and they need these tools to be able to work together -- not just the ones that Rational is building, but the ones that everyone's constructing. So, this focus on integration and making it so that we can really in a seamless way work from end to end.

The next is the need to measure and reflect on the way that the work's being done. And we call this the Measured Capability Improvement Framework, but it's really about improved process. It's good to have a process, but one really needs to understand how that process is impacting the productivity of the teams and the effectiveness of the teams by establishing the right set of measurements and then

adapting the process to actually get the desired results.

So this [INAUDIBLE] governance is something that is really maturing over the last few years. We saw it, just hints of it initially, and really focused on process enactment, but now a much more serious effort to enable increased productivity.

And the final thought is an ongoing effort in making reductions in the cost of ownership. So things that make these, administration of the products that we provide and the...[being able] to use multiple products together, is something that we've always talked about and may not sound super exciting, but provides huge value to everyone involved. So I would say that would be the big issues: integration, measurement and process, and reductions in the cost of ownership.

MATHENY: So, John, how has Jazz evolved from an architectural perspective?

WIEGAND: So with these strategies that we've put together, if we look at what we're doing architecturally, we've always been aware of the importance of the Web. In fact, we've had a Web client from the beginning that we've been using.

And I would say our transition architecturally has been from being Web aware to being Web centric and saying we will work around the architecture of the Web and have that just permeate the way we do things: leverage the scalability characteristics the Web has and the distributed nature; the reliability that we all appreciate in the Web...

And the extensibility, the fact that you can do things and the very simple things are simple, and that you can navigate from place to place and it's predictable, things work in an expected way.

So that architectural evolution has really...went deep into the core of what we do with Jazz, and now we have this loosely coupled yet rich experience that we provide with our Jazz products. We can go from Rational Quality Manager, navigate to Rational Team Concert, do that navigation, and from a user experience point of view, it appears that you stayed in one product, just saw a widget come up that described the transition, but you felt like you were in one seamless environment.

That was the vision we had before, and now we're delivering it. And not just navigating between a set of two Rational products there, but any products that would implement the correct interfaces. So we have this architecture for integration, which makes all this stuff possible.

MATHENY: And Kartik, in your work with customers, what are some of the most common misunderstandings about the Jazz platform?

KANAKASABESAN: Again, a very good question there, Angelique. Just working with customers -- and I think this is also partly self inflicted -- was the initial availability of Jazz was in the form of Rational Team Concert. And what customers kind of assumed was the fact that Rational Team Concert was analogous to Jazz in and of itself.

And there was a lot of work that was done to kind of separate the two, but for all well intentions that we had, the download of the consumability of Jazz was still through Rational Team Concert until Requirements Composer and Quality Manager kind of became real.

So to a very large extent, until Rational Team Concert 2.0 came out, the assumption was the fact that Jazz platform was Rational Team Concert, and Rational Team Concert was the Jazz platform.

And what John has been alluding to as well was the fact that when we talk about in the context of the Jazz platform in general, it's encompassing a collaborative ALM solution

architecture, if you will, and encompasses both Team Concert, R2M and Composer, and potentially other tools that will join or will be rebuilt on this particular platform going forward as well.

So the common misconception that customers usually have is the fact that Rational Team Concert, or Rational Quality Manager, or Requirements Composer, are manifestations of Jazz, where in reality they're proof points of the integration architecture in general.

A You know, Kartik, you just mentioned these products -- Rational Team Concert, Quality Manager and the Requirements Composer. They comprise the first wave of Jazz products. What was early customer feedback on these products?

KANAKASABESAN: Truth be told, the feedback that we've gotten has been phenomenal. You know, we were completely taken aback with the way the community has actually gone ahead and embraced these tools, and also given us very constructive feedback. As a matter of fact, when the first set of tools that came out, the customer base took these tools up, [asked] it was very constructive feedback on how we could improve the experience.

And to that effect, we took those feedback to heart and

delivered on our CALM -- our Collaborative ALM vision, which we call the CLM 1.0 -- where in which we are now showcasing integrations between these tools in a very seamless fashion.

So the feedback has been very energetic in general, and we've gotten great feedback, we've gotten good feedback. We've also gotten not so good feedback. But all in all, it's all positive and it's actually helping us improve the way we build and deliver our products and actually consume them even internally, for that matter.

A And John, what are some of the key services that the Jazz platform provides to the products that leverage it?

WIEGAND: When people think of the services the Jazz platform provides, the first things that come to mind are storage -- how do you persist the data that an applications wants to store, query, and get it back and combine it in new and interesting ways.

And process services, ways of putting control on the access to the capability and be able to put preconditions and follow up actions around the specific operations that are being done. And that's a good general introduction to the kind of services that are available.

I also like to go talk about the integration building blocks that we provide at the same time, because that's just as critical for teams that are building products to a platform -- capabilities like presentation services to enable the construction of Web UIs, widgets to work with to support navigation and to do rich helpers. So, these are the kind of capabilities that enable someone to get started and to use the Jazz technology that you need to build everything from the ground up.

A And dashboards and reporting capabilities are part of the Jazz platform and included in every product. John, what's the value add that customers get with Rational Insight?

WIEGAND: That's another good question. So, the reporting capability that every product has and the dashboarding there is valuable for exposing the capabilities of that individual product. But what Insight brings to the table is the ability to do cross product reporting, and to do analysis across the information that came from these disparate sources, bringing all that data together and now being in in a position to do analysis.

We like to call that development intelligence: doing the analysis on the disparate data that's been brought together and being able to make conclusions that you couldn't do

before. For example, being able to understand the degree of test coverage as it matches with the build success. So that's a correlation you can do when you bring both pieces of information together that you couldn't do independently.

MATHENY: So, how will Jazz technology evolve with respect to asset management?

WIEGAND: That's another good domain to explore. Asset management is an integral part of the lifecycle. We don't start doing development from scratch; we have assets in hand that we can work with. When we're done doing our development effort, we then publish those resources and make them available for subsequent efforts.

In fact, we have assets like this that show up all the time. We have requirements templates that we use, we have test cases that we use. We bring those along from effort to effort. So asset management is a fundamental part of Collaborative ALM. So that's the reason it needs to be part of the story, and it is part of our ongoing efforts.

There's other synergies that occur between Jazz technology and asset management, and that is, we need approval flows around the assets. We need to have discussions around the assets -- the approval flows deciding whether they're ready to share, discussions to exploring a range of

characteristics about the individual assets. Again, all of these bring the notion of asset management and make it an integral part of the Jazz solution.

MATHENY: And Kartik, social software is increasingly a core capability of software products. How will the Jazz platform evolve with respect to social software?

KANAKASABESAN: Again, another good question there. So, social software, as you know, with Web 2.0 being as pervasive as it is, the power of providing content is no longer available to a select few. It's actually now with everyone who has a mobile device or anything of that kind are actually power content providers.

And we have some software products that are actually doing very good at that. We have instances where for example within just the IBM Software Group we have, you know, Lotus Connections as a capability to bring social software kind of like capabilities into the enterprise.

And what that basically means is, when we talk about social software in that context, it's bringing the notion of things like Facebook into the organization. And how that helps the organization in general is the fact that you're able to manage knowledge effectively, be able to create just in time teams that need to respond to a changing business need.

Now, when you look at it in the context of social software and see how that impacts software delivery, they're pretty much alike. With the advent of Agile methodologies and so on and so forth, people need dynamic teams. When there are teams that are distributed, you need to be able to create these skills and access people with the right skills at the right time to respond to a changing business need.

So, there's work that's currently actively happening in the context of Jazz and Jazz products specifically, working closely with products like Lotus Connections and Microsoft Sharepoint services as well, to be able to harness the social software capability and bring that in into software development practices across the application lifecycle.

MATHENY: So, Kartik, can you explain the configuration strategy for customers who wish to purchase one or more Jazz products?

KANAKASABESAN: So, going back to what we did in the past. When the first wave of Jazz products came out, the typical configuration was each instance of the application would run on its own little server. Again, feedback, we heard our customers and said, okay, you know, this is going back to the same old problem that we had with some of the other tools; this has got to be done better.

And at the end of the day we agree to the fact that putting everything in a single repository is not probably the right way of dealing with things effectively, but the fact that being able to link and plug into resources, and again, back to the architecture statement that John alluded to earlier, is the right way to go...

We've actually made some significant improvements in the Jazz offerings where in which now they will be able to share single instances of application servers, or a single database server to be able to go ahead and store all the databases of these respective products...

But still, you don't lose out the linking capability which makes the experience seem like it's all a single product. So there's been work that's actually happened in that context where in which customers when they purchase these products, they could deploy it in various fashions.

You could still have all these tools at a central location with teams around the world plugging into it. Or, depending on functional divisions within the organization -- so for example, if you had the development team in North America, you had the business analysts in Europe, and let's say the testing team in China. And they all had the respective servers installed there for their respective domains, you

still want to be able to link and trace artifacts across these different servers.

So, whether it's all co-located or separated across based on functional boundaries, the user experience still remains consistent and seamless as if they were all working together as one.

And those are the kind of things that we've gone ahead and done with Jazz, and that's what customers, when purchasing these products -- whether it be, you know, standalone products or more -- will actually go and experience.

MATHENY: John, what role do open source, third party and partner products play in the Jazz strategy?

WIEGAND: These products -- the open source products, the third party products, partner products -- all these kinds of products are integral and instrumental in the Jazz strategy.

If you look at it from the customer point of view, the customer has a wide range of products from all these sources. In fact, some of the products that the customer has are homegrown, internally developed.

But the integration requirements are there just like they are with the other Jazz products. So we treat all of these as an integral part of the story and a necessary part, and a

vibrant ecosystem of products that integrate is essential to our success as an industry.

So what do we need to do about that? We need an open architecture that's inclusive, and then we need to do things that allow integration to occur without a significant adoption cost.

So we've established an industry initiative, it's called Open Services for Lifecycle Collaboration. We often shorten that to open services or OSLC. And the intent of that initiative is to look at how we can make it easier to use lifecycle tools, and in fact to use lifecycle tools together. The focus isn't on an individual tool, but it's on how these tools come together and are effective in concert.

So what's going on at open services? We're defining specifications for the resources: what is a requirement? What is a defect? We understand what those resources are and we provide a common format for them. And then we define some services for working with them. And by establishing these common interfaces, we're using the Web architecture again, these are Restful interfaces...

By having those interfaces in hand, we can now do integrations between products implemented in any technology

beneath the seams. So we now have enabled it for a broad ecosystem, whether it's open source, third party or other, to tap into this integration style, taking the technologies they have and integrating them.

We've already seen some success in this where the Eclipse Modeling Project has built a client that integrates with change management systems that support the open services interface. So that provides integration with ClearQuest, with Rational Team Concert and more products that will support this interface over time.

So, as more and more organizations and individuals get involved in the open services initiative, we look at seeing a broad cross section of interfaces across many domains, and we see a lot of potential there.

Now, Jazz is going to add additional value atop open services. We've already talked about the additional services and building blocks that Jazz provides, provides an SDK as well to help someone in the construction of their application.

And of course, an implementation of the Jazz Team Server. So the combination of open services providing common interfaces and common specifications with Jazz we think is a winning combination for all.

MATHENY: And Kartik, our last question today. What's the best way for customers and our audience to learn more about the Jazz platform and Jazz products?

KANAKASABESAN: Well, I would say, in a very simple answer would be jazz.net, just like the way I would expect our users who are listening to the podcast here to be able to consume this either through IBM developerWorks or through jazz.net.

But a lot of what we have spoken about so far is available through jazz.net as the primary medium of communication with the larger community. You know, if we were to go on to jazz.net today, there is information there about what we plan to go ahead and deliver. So we've been doing that for the last two years, and there's no way that we're going to stop now. This information is available with a searchable infrastructure as it is.

There's been a lot of work to that end that's been done on jazz.net as well. As a matter of fact, if the community members were to go on to jazz.net, look at it the way it is now versus how they had experienced it in the past, there's some significant improvements.

Along with that, just to kind of emphasize, all the products

that we've mentioned so far in our podcast here, the information regarding that is also available on jazz.net. So, more product teams within Rational are adopting this open, transparent model to actually reach out to the customers and solicit feedback a lot early than they were in the past and actually be able to deliver value a lot quicker to those customers as well.

So, not only will they be able to learn, but also participate in evolution of these technologies that we've been talking about in the context of Jazz.

MATHENY: So jazz.net it is. John and Kartik, thank you so much for sharing your time today to discuss what's happening with Jazz. We really appreciate it.

WIEGAND: Angelique, I enjoyed this conversation. Thanks for setting this up.

KANAKASABESAN: Yes, same here. Same goes here, Angelique. Thank you very much for setting this up.

MATHENY: That's great.

That was Rational Chief Architect, John Wiegand, and Jazz Jumpstart Manager, Kartik Kanakasabesan. If you are interested in more podcasts like this one, check out the Rational Talks To You Podcast Page at www.ibm.com/rational/podcasts.

What's happening with Jazz? Angelique Matheny, with John Wiegand, Kartik Kanakasabesan

We'll include the link to the teleconference, Working Smarter Enriching the Jazz Environment with Social Software.

Be sure to check it out. This has been an IBM podcast.

I'm Angelique Matheny. Thanks for listening. Keep tuning in as Rational Talks To You.

IBM Podcast

[MUSIC] [END OF SEGMENT]