



PODCAST

To listen to the entire interview,
log on to www.btquarterly.com

building YOUR CLOUD

Service-oriented architecture (SOA) is being used across many enterprises today, but implementation strategies are extremely varied. For examples of how it's being used well, *BTQ* talks with **CRAIG MUZILLA**, Vice President of JBoss Enterprise Middleware at Red Hat, Inc.

Business Trends Quarterly To frame our discussion a bit, what is your definition of SOA?

CRAIG MUZILLA SOA is a new modern architectural approach for building and integrating applications. Essentially, it provides a great degree of flexibility in how one can create applications. SOA is much easier and simpler than the traditional means for creating applications, and much more dynamic.

It's based on a concept called abstraction. What abstraction means is that we will create services that are really business processes – it could be taking an order or looking up a customer profile – and we're hiding the underlining implementation of how that code's written. (The code) could be written a number of different ways, but as long as the interface is abstracting the implementation, it makes it much easier to use that business process and can be reused over and over. Also, if the implementation changes, it won't break the application, making it much more dynamic for putting together applications and integrating processes across an organization or even across boundaries.

BTQ Some of the things you discussed, such as the reuse aspect, seem like they could be important to enterprise IT. Can you touch on that? In other words, what is SOA's value proposition?

CM There are quite a few benefits to SOA. First, it provides a high degree of interoperability between business processes and applications. It makes it much easier to integrate things together. For instance, a company might have a financial system, customer relations management system

and suppliers. SOA makes it much easier to tie all those systems together and... respond to business changes. So, if there's a new process that needs to be incorporated into an application, or a new workflow, such as a new way of doing claims processing, it's very easy to make those changes. With traditional software, you'd have to rip the software out or rewrite the code, which is very difficult to do.

SOA also allows a company to integrate a number of diverse systems, like Web-based applications that may have been written in the Java language or older systems that were written on a mainframe. SOA makes it much easier to bring those (systems) together. Finally, there is a lot of reuse of processes and more consistencies. For instance, defining what a customer is for a company. In the past, if you had three or four different applications, customer data was replicated across these three or four applications. Now you can create a customer service used across all your applications, reducing the cost of having to maintain the customer data while providing a great level of consistency.

BTQ Can you discuss Red Hat's version of SOA and its strategy, as well as where enterprise middleware (software) comes in?

CM Red Hat is an open source infrastructure software company. We provide operating system software in the form of Linux, virtualization software, and middleware software, which is part of our JBoss brand. Our strategy as it relates to SOA is to provide a whole portfolio of middleware products that help companies implement SOA.

We provide middleware products that help you build these types of applications, and to deploy, run, integrate and manage them. Among the types of products that we offer, we have a world class application server and platform, development tools, an enterprise service bus, which is some of the fundamental plumbing in our SOA environment, business process management software and software for creating Web 2.0 applications. All of these are provided to help a company implement an SOA-based approach to how they're developing and integrating their applications.

BTQ How does SOA help companies with cloud computing?

CM One could easily argue that in many ways, cloud computing isn't possible without SOA. Through SOA, one can more readily move business processes into the cloud and more readily use these processes. No longer do all of your processes need to be run and owned locally into siloed applications. For example, SOA makes it possible for SaaS providers to offer business services that can be integrated with a company's own processes. These external services could be things such as logistics tracking, tax calculations, currency exchange and many more. By service-enabling these processes, companies don't need to write their own code – they can leverage these services into composite applications. Also, companies can more readily create a business process and push it into the cloud for greater utilization, but through SOA, still make is possible to access the process regardless of its location.

Red Hat JBoss products, such as our

application platforms, enterprise service bus and business process management software, make it easy to service processes for the cloud. JBoss can help a company utilize external cloud-based services and then orchestrate a collection of cloud services into a comprehensive business application or set of processes.

BTQ What is unique about Red Hat's approach?

CM Probably the most unique thing about our approach is that everything is open source. Open source software means that when we deliver a product to a customer, they not only have the binary code, but they also have the source code and can see everything that is going on underneath the covers. Everything is built in a community process, so if customers want to contribute code, they can do so. They can also see what's coming down the road in terms of new products that are being developed. The approach we've taken with the JBoss portfolio is that since products have all been developed in the community, and we've helped drive the community, everything works together.

When companies implement SOA, using middleware pieces to do that, they may have different pieces (of software) even within the same company and often these pieces don't work together smoothly. We've architected everything to work together well and we're the only vendor in the marketplace that's providing all the pieces under an open source model. What does open source really provide to a business? There's a lot of transparency and flexibility, especially in an SOA environment, and fundamentally, it's a much more cost effective way to not only do the development, but also run these applications over time.

BTQ Could you give an example where a company is using your version of SOA well?

CM Let me give you a couple of examples. The first one is a customer in Europe. The Swedish railroad has implemented many of our products within the JBoss portfolio for an SOA-based application. They were trying to develop a more flexible software delivery system in their company and their first application implemented was a ticket auctioning system. Often the railroad will have an excess of tickets during the day, seats that go unfilled. What they wanted to do was be able to post those tickets on a subsidiary of eBay and sell them to the highest bidder so they could fully sell the seats. This required a level of interoperability between

the outside provider that would be hosting these tickets and the Swedish railroad themselves. Using a traditional means of development for this would have been very complex, very costly and very brittle. If anything would have changed, it would have broken the application. Using an SOA-based approach, and JBoss products from Red Hat, the Swedish railroad was very easily able to implement this ticket auctioning system, integrate with the eBay subsidiary to sell the tickets, and they did it in four months rather than the estimated twelve months it would have taken with traditional means. They were also able to get a very fast return on investment (ROI) and sell an additional 2,000 tickets a week.

The second example is a company called Toronto Hydro, a utility company based in Toronto, Canada. They were implementing an SOA-based approach for all their systems. One of the first applications was something called a Smart Meter system. They needed a way... to tie their SAP system and Oracle financial system together and supply information about the electricity being used by a consumer so (the data) could be introduced to a variable-based pricing model... Since these systems would often change, and some of the underlying implementation would therefore change, SOA was the best way to go. Toronto Hydro used the JBoss products to tie the systems together and then offer this Smart Meter system to their consumers.

BTQ For both of these implementations, it sounds like there were some major changes that were taking place during the switch to SOA and the JBoss suite. What were some of the pain points experienced and how were they solved?

CM Using a more traditional approach, it would have been very difficult to have the level of integration of the various

processes, both inside and outside these two companies. By using an SOA-based approach and the JBoss product line, it was easy to create service interfaces for all these business processes, couple them together and... (make it) easy for developers to implement and use. It saved a considerable amount of time.

At the end of their projects, once they had these two applications up and running, the companies can now be much more flexible. If there was anything they needed to change, such as the underlying implementation, they can make those changes without breaking the application and they can diagnose any issues because of the open source nature of the JBoss product line. Their interface with the JBoss community and through Red Hat also allows the companies to offer suggestions for changes to new software releases and new projects.

BTQ What were some of the benefits that companies gained from making this change to SOA and JBoss products?

CM It was a faster time to implement under both scenarios. In general, using JBoss and the concept of SOA makes it faster to implement rather than having to hard code products together. Because they were faster, their time to ROI was much faster. One of the big benefits, coming back to flexibility, was that if any of these processes need to change in the future, companies can make these changes to applications easily without having to rewrite the initial implementation. No business is static, they are always changing... and because of the JBoss product line and SOA, a company can be much more agile. They'll be able to change these applications as they gain more experience with their customers or as market conditions change. It becomes a much more dynamic way of executing these two capabilities. **BTQ**



CRAIG MUZILLA
VICE PRESIDENT, MIDDLEWARE BUSINESS UNIT
RED HAT, INC.

Craig Muzilla is responsible for Red Hat's overall middleware and JBoss product business. Mr. Muzilla provided leadership establishing early Web services industry initiatives and has led strategy, product management, marketing and business development for many middleware software companies. Prior to Red Hat, he was Senior Vice President at MetaMatrix, a leader in data services middleware that was acquired by Red Hat.

